

A. ASSEMBLY DO ATMEGA

A.1 INSTRUÇÕES DO ATMEGA

Mnemônico	Operandos	Descrição	Operação	Flags	Clocks
INSTRUÇÕES ARITMÉTICAS E LÓGICAS					
ADD	Rd, Rr	Soma dois registradores	$Rd \leftarrow Rd + Rr$	Z, C, N, V, H	1
ADC	Rd, Rr	Soma dois registradores com <i>Carry</i>	$Rd \leftarrow Rd + Rr + C$	Z, C, N, V, H	1
ADIW	Rdl, K	Soma o valor imediato à palavra (16 bits)	$Rdh:Rdl \leftarrow Rdh:Rdl + K$	Z, C, N, V, S	2
SUB	Rd, Rr	Subtrai dois registradores	$Rd \leftarrow Rd - Rr$	Z, C, N, V, H	1
SUBI	Rd, K	Subtrai uma constante do registrador	$Rd \leftarrow Rd - K$	Z, C, N, V, H	1
SBC	Rd, Rr	Subtrai dois registradores com <i>Carry</i>	$Rd \leftarrow Rd - Rr - C$	Z, C, N, V, H	1
SBCI	Rd, K	Subtrai constante de registrador com <i>Carry</i>	$Rd \leftarrow Rd - K - C$	Z, C, N, V, H	1
SBIW	Rdl, K	Subtrai valor imediato da palavra (16 bits)	$Rdh:Rdl \leftarrow Rdh:Rdl - K$	Z, C, N, V, S	2
AND	Rd, Rr	Lógica E entre registradores	$Rd \leftarrow Rd \bullet Rr$	Z, N, V	1
ANDI	Rd, K	Lógica E entre registrador e constante	$Rd \leftarrow Rd \bullet K$	Z, N, V	1
OR	Rd, Rr	Lógica OU entre registradores	$Rd \leftarrow Rd \vee Rr$	Z, N, V	1
ORI	Rd, K	Lógica OU entre registrador e constante	$Rd \leftarrow Rd \vee K$	Z, N, V	1
EOR	Rd, Rr	OU EXCLUSIVO entre registradores	$Rd \leftarrow Rd \otimes Rr$	Z, N, V	1
COM	Rd	Complemento de um	$Rd \leftarrow 0xFF - Rd$	Z, C, N, V	1
NEG	Rd	Complemento de dois	$Rd \leftarrow 0x00 - Rd$	Z, C, N, V, H	1
SBR	Rd, K	Ativa bit(s) no registrador	$Rd \leftarrow Rd \vee K$	Z, N, V	1
CBR	Rd, K	Limpa bit(s) no registrador	$Rd \leftarrow Rd \bullet (0xFF - K)$	Z, N, V	1
INC	Rd	Incrementa registrador	$Rd \leftarrow Rd + 1$	Z, N, V	1
DEC	Rd	Decrementa registrador	$Rd \leftarrow Rd - 1$	Z, N, V	1
TST	Rd	Teste de zero ou negativo	$Rd \leftarrow Rd \bullet Rr$	Z, N, V	1
CLR	Rd	Limpa registrador	$Rd \leftarrow Rd \otimes Rd$	Z, N, V	1
SER	Rd	Ativa registrador (todos os bits em 1)	$Rd \leftarrow 0xFF$	Nenhum	1
MUL	Rd, Rr	Multiplica sem sinal	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
MULS	Rd, Rr	Multiplica com sinal	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
MULSU	Rd, Rr	Multiplica registrador com sinal e sem sinal	$R1:R0 \leftarrow Rd \times Rr$	Z, C	2
FMUL	Rd, Rr	Multiplicação fracionária sem sinal	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
FMULS	Rd, Rr	Multiplicação fracionária com sinal	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
FMULSU	Rd, Rr	Multiplicação fracionária de registrador com sinal e sem sinal	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z, C	2

INSTRUÇÕES DE DESVIO					
RJMP	k	Desvio relativo	$PC \leftarrow PC + k + 1$	Nenhum	2
IJMP		Desvio indireto para (Z)	$PC \leftarrow Z$	Nenhum	2
RCALL	k	Chama de sub-rotina	$PC \leftarrow PC + k + 1$	Nenhum	3
ICALL		Chamada indireta para (Z)	$PC \leftarrow Z$	Nenhum	3
RET		Retorno de sub-rotina	$PC \leftarrow STACK$	Nenhum	4
RETI		Retorno de interrupção	$PC \leftarrow STACK$	I	4
CPSE	Rd, Rr	Compara, pula se igual	$\text{if}(Rd=Rr) PC \leftarrow PC + 2 \text{ ou } 3$	Nenhum	1/ 2/ 3
CP	Rd, Rr	Compara	$Rd - Rr$	Z,N,V,C,H	1
CPC	Rd, Rr	Compara com Carry	$Rd - Rr - C$	Z,N,V,C,H	1
CPI	Rd, K	Compara registrador com valor imediato	$Rd - K$	Z,N,V,C,H	1
SBRC	Rr, b	Pula se o bit do registrador estiver limpo (0)	$\text{if}(Rr(b)=0) PC \leftarrow PC + 2 \text{ ou } 3$	Nenhum	1/ 2/ 3
SBRs	Rr, b	Pula se o bit do registrador estiver ativo (1)	$\text{if}(Rr(b)=1) PC \leftarrow PC + 2 \text{ ou } 3$	Nenhum	1/ 2/ 3
SBIC	P, b	Pula se o bit do registrador de I/O estiver limpo (0)	$\text{if}(P(b)=0) PC \leftarrow PC + 2 \text{ ou } 3$	Nenhum	1/ 2/ 3
SBIS	P, b	Pula se o bit do registrador de I/O estiver ativo (1)	$\text{if}(P(b)=1) PC \leftarrow PC + 2 \text{ ou } 3$	Nenhum	1/ 2/ 3
BRBS	s, k	Desvia se o bit de sinalização de status estiver ativo (1)	$\text{if}(SREG(s)=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRBC	s, k	Desvia se o bit de sinalização de status estiver limpo (0)	$\text{if}(SREG(s)=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BREQ	k	Desvia se igual	$\text{if}(Z=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRNE	k	Desvia se diferente	$\text{if}(Z=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRCS	k	Desvia se o bit de Carry estiver ativo (1)	$\text{if}(C=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRCC	k	Desvia se o bit de Carry estiver limpo (0)	$\text{if}(C=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRSH	k	Desvia se igual ou maior	$\text{if}(C=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRLO	k	Desvia se menor	$\text{if}(C=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRMI	k	Desvia se negativo	$\text{if}(N=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRPL	k	Desvia se positivo	$\text{if}(N=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRGE	k	Desvia se maior ou igual, com sinal	$\text{if}(N \otimes V = 0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRLT	k	Desvia se menor que zero, com sinal	$\text{if}(N \otimes V = 1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRHS	k	Desvia se o bit sinalizador de Carry auxiliar estiver ativo (1)	$\text{if}(H=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRHC	k	Desvia se o bit sinalizador de Carry auxiliar estiver limpo (0)	$\text{if}(H=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRTS	k	Desvia se o bit sinalizador T estiver ativo (1)	$\text{if}(T=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRTC	k	Desvia se o bit sinalizador T estiver limpo (0)	$\text{if}(T=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRVS	k	Desvia se o bit sinalizador de estouro estiver ativo (1)	$\text{if}(V=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRVC	k	Desvia se o bit sinalizador de estouro estiver limpo (0)	$\text{if}(V=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRIE	k	Desvia se a interrupção estiver habilitada	$\text{if}(I=1) PC \leftarrow PC + k + 1$	Nenhum	1/ 2
BRID	k	Desvia se a interrupção estiver desabilitada	$\text{if}(I=0) PC \leftarrow PC + k + 1$	Nenhum	1/ 2

INSTRUÇÕES PARA A TRANSFERÊNCIA DE DADOS

MOV	Rd, Rr	Movimento entre registradores	$Rd \leftarrow Rr$	Nenhum	1
MOVW	Rd, Rr	Copia registrador de palavra	$Rd + 1:Rd \leftarrow Rr + 1:Rr$	Nenhum	1
LDI	Rd, K	Carrega valor imediato	$Rd \leftarrow K$	Nenhum	1
LD	Rd, X	Carrega indiretamente	$Rd \leftarrow (X)$	Nenhum	2
LD	Rd, X+	Carrega indiretamente com pós-incremento	$Rd \leftarrow (X), X \leftarrow X + 1$	Nenhum	2
LD	Rd, -X	Carrega indiretamente com pré-decremento	$X \leftarrow X - 1, Rd \leftarrow (X)$	Nenhum	2
LD	Rd, Y	Carrega indiretamente	$Rd \leftarrow (Y)$	Nenhum	2
LD	Rd, Y+	Carrega indiretamente com pós-incremento	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	Nenhum	2
LD	Rd, -Y	Carrega indiretamente com pré-decremento	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	Nenhum	2
LDD	Rd, Y+q	Carrega indiretamente com deslocamento	$Rd \leftarrow (Y + q)$	Nenhum	2
LD	Rd, Z	Carrega indiretamente	$Rd \leftarrow (Z)$	Nenhum	2
LD	Rd, Z+	Carrega indiretamente com pós-incremento	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	Nenhum	2
LD	Rd, -Z	Carrega indiretamente com pré-decremento	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	Nenhum	2
LDD	Rd, Z+q	Carrega indiretamente com deslocamento	$Rd \leftarrow (Z + q)$	Nenhum	2
LDS	Rd, k	Carrega diretamente da SRAM	$Rd \leftarrow (k)$	Nenhum	2
ST	X, Rr	Armazena indiretamente	$(X) \leftarrow Rr$	Nenhum	2
ST	X+, Rr	Armazena indiretamente com pós-incremento	$(X) \leftarrow Rr, X \leftarrow X + 1$	Nenhum	2
ST	-X, Rr	Armazena indiretamente com pré-incremento	$X \leftarrow X - 1, (X) \leftarrow Rr$	Nenhum	2
ST	Y, Rr	Armazena indiretamente	$(Y) \leftarrow Rr$	Nenhum	2
ST	Y+, Rr	Armazena indiretamente com pós-incremento	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	Nenhum	2
ST	-Y, Rr	Armazena indiretamente com pré-incremento	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	Nenhum	2
STD	Y+q, Rr	Carrega indiretamente com deslocamento	$(Y + q) \leftarrow Rr$	Nenhum	2
ST	Z, Rr	Armazena indiretamente	$(Z) \leftarrow Rr$	Nenhum	2
ST	Z+, Rr	Armazena indiretamente com pós-incremento	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	Nenhum	2
ST	-Z, Rr	Armazena indiretamente com pré-incremento	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	Nenhum	2
STD	Z+q, Rr	Carrega indiretamente com deslocamento	$(Z + q) \leftarrow Rr$	Nenhum	2
STS	k, Rr	Carrega diretamente para SRAM	$(k) \leftarrow Rr$	Nenhum	2
LPM		Carrega a memória de programa	$R0 \leftarrow (Z)$	Nenhum	3
LPM	Rd, Z	Carrega a memória de programa	$Rd \leftarrow (Z)$	Nenhum	3
LPM	Rd, Z+	Carrega a memória de programa com pós-incremento	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	Nenhum	3
SPM		Armazena a memória de programa	$(Z) \leftarrow R1:R0$	Nenhum	-
IN	Rd, P	Leitura de registrador de I/O	$Rd \leftarrow P$	Nenhum	1
OUT	P, Rr	Escrita de registrador de I/O	$P \leftarrow Rr$	Nenhum	1
PUSH	Rr	Coloca registrador na pilha (Stack)	$STACK \leftarrow Rr$	Nenhum	2
POP	Rd	Retira registrador da pilha (Stack)	$Rd \leftarrow STACK$	Nenhum	2

INSTRUÇÕES DE BIT E TESTE DE BIT

SBI	P, b	Ativa o bit no registrador de I/O	$I/O(P,b) \leftarrow 1$	Nenhum	2
CBI	P, b	Limpa o bit do registrador de I/O	$I/O(P,b) \leftarrow 0$	Nenhum	2
LSL	Rd	Deslocamento lógico à esquerda	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	Z, C, N, V	1
LSR	Rd	Deslocamento lógico à direita	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	Z, C, N, V	1
ROL	Rd	Rotação à esquerda através do Carry	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	Z, C, N, V	1
ROR	Rd	Rotação à direita através do Carry	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	Z, C, N, V	1
ASR	Rd	Deslocamento aritmético à direita	$Rd(n) \leftarrow Rd(n+1), n=0..6$	Z, C, N, V	1
SWAP	Rd	Troca nibbles (4 bits)	$Rd(3..0) \leftarrow Rd(7..4), Rd(7..4) \leftarrow Rd(3..0)$	Nenhum	1
BSET	s	Ativa bit (flag)	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Limpa bit (flag)	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr, b	Armazenagem do bit do registrador para o T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Carga do bit T para o registrador	$Rd(b) \leftarrow T$	Nenhum	1
SEC		Ativa o Carry	$C \leftarrow 1$	C	1
CLC		Limpa o Carry	$C \leftarrow 0$	C	1
SEN		Ativa o bit de negativo	$N \leftarrow 1$	N	1
CLN		Limpa o bit de negativo	$N \leftarrow 0$	N	1
SEZ		Ativa o bit de zero	$Z \leftarrow 1$	Z	1
CLZ		Limpa o bit de zero	$Z \leftarrow 0$	Z	1
SEI		Habilita a interrupção global	$I \leftarrow 1$	I	1
CLI		Desabilita interrupção global	$I \leftarrow 0$	I	1
SES		Ativa o bit de teste de sinal	$S \leftarrow 1$	S	1
CLS		Desativa o bit de teste de sinal	$S \leftarrow 0$	S	1
SEV		Ativa o bit de estouro do complemento de dois	$V \leftarrow 1$	V	1
CLV		Limpa o bit de estouro do complemento de dois	$V \leftarrow 0$	V	1
SET		Ativa o bit T	$T \leftarrow 1$	T	1
CLT		Limpa o bit T	$T \leftarrow 0$	T	1
SEH		Ativa o bit de Carry auxiliar	$H \leftarrow 1$	H	1
CLH		Limpa o bit de Carry auxiliar	$H \leftarrow 0$	H	1

INSTRUÇÕES DE CONTROLE DA UNIDADE MICROCONTROLADA

NOP		Nenhuma operação		Nenhum	1
SLEEP		Entra no modo sleep	Consultar manual do fabricante	Nenhum	1
WDR		Reset do Watchdog	Consultar manual do fabricante	Nenhum	1

- Ativação de bit significa colocá-lo em 1.
- Limpeza de bit significa colocá-lo em 0.

Legenda:

Rd	= registrador de destino (e origem).
Rr	= registrador de origem.
s, b	= constante (0..7); pode ser uma expressão constante.
K	= constante, faixa de valores depende da instrução; pode ser uma expressão constante.
P	= constante (0..31/63), pode ser uma expressão constante ou um registrador de I/O.
q	= constante (0..63), pode ser uma expressão constante.
Rdl	= R24, R26, R28, R30, usado nas instruções ADIW e SBIW.
X, Y, Z	= registradores de 16 bits para endereçamento indireto (X=R27:26, Y=R29:28 e Z=R31:30).

Todas as instruções que operam com registradores de uso geral tem acesso direto em um único ciclo a todos eles. As exceções são as cinco instruções lógicas e aritméticas entre uma constante e um registrador, SBCI, SUBI, CPI, ANDI e ORI, e a instrução para carga de constante imediata, LDI. Essas instruções se aplicam somente a metade superior dos registradores de uso geral (R16..R31). As instruções SBC, SUB, CP, AND, OR e as demais operações entre um ou dois registradores se aplicam a todo o banco de registradores.