

Μνημονικό	Ορίσματα	Περιγραφή	Αντιστοίχια	Σημασίες	Κόστος
-----------	----------	-----------	-------------	----------	--------

Εντολές μεταφορής δεδομένων

MOV	Rd, Rt	Move Between Registers	Rd← Rt	-	1
MOVW	Rd, Rt	Copy Register Word	Rd+1:Rd← Rt+1:Rt	-	1
LDI	Rd, K	Load Immediate	Rd ← K	-	1
LD	Rd, X	Load Indirect	Rd ← (X)	-	2
LD	Rd, X+	Load Indirect and Post-Inc.	Rd ← (X), X← X+1	-	2
LD	Rd, - X	Load Indirect and Pre-Dec.	X← X - 1, Rd← (X)	-	2
LD	Rd, Y	Load Indirect	Rd ← (Y)	-	2
LD	Rd, Y+	Load Indirect and Post-Inc.	Rd ← (Y), Y← Y+1	-	2
LD	Rd, - Y	Load Indirect and Pre-Dec.	Y← Y - 1, Rd← (Y)	-	2
LDD	Rd, Y+q	Load Indirect + Displacement	Rd ← (Y+q)	-	2
LD	Rd, Z	Load Indirect	Rd ← (Z)	-	2
LD	Rd, Z+	Load Indirect and Post-Inc.	Rd ← (Z), Z← Z+1	-	2
LD	Rd, - Z	Load Indirect and Pre-Dec.	Z← Z - 1, Rd← (Z)	-	2
LDD	Rd, Z+q	Load Indirect + Displacement	Rd ← (Z+q)	-	2
LDS	Rd, k	Load Direct from SRAM	Rd ← (k)	-	2
ST	X, Rt	Store Indirect	(X)← Rt	-	2
ST	X+, Rt	Store Indirect and Post-Inc.	X)← Rt, X← X+1	-	2
ST	- X, Rt	Store Indirect and Pre-Dec.	X ← X - 1, (X)← Rt	-	2
ST	Y, Rt	Store Indirect	(Y)← Rt	-	2
ST	Y+, Rt	Store Indirect and Post-Inc.	(Y)← Rt, Y← Y+1	-	2
ST	- Y, Rt	Store Indirect and Pre-Dec.	Y ← Y - 1, (Y)← Rt	-	2
STD	Y+q,Rt	Store Indirect + Displacement	(Y+q) ← Rt	-	2
ST	Z, Rt	Store Indirect	(Z)←Rt	-	2
ST	Z+, Rt	Store Indirect and Post-Inc.	(Z)←Rt, Z← Z+1	-	2
ST	- Z, Rt	Store Indirect and Pre-Dec.	Z ← Z - 1, (Z)←Rt	-	2
STD	Z+q,Rt	Store Indirect + Displacement	(Z+q) ← Rt	-	2
STS	k, Rt	Store Direct to SRAM	(k) ← Rt	-	2
LPM		Load Program Memory	R0 ← (Z)	-	3
LPM	Rd, Z	Load Program Memory	Rd ← (Z)	-	3
LPM	Rd, Z+	Load Program Memory	Rd ← (Z), Z← Z+1	-	3
SPM		Store Program Memory	(Z) ← R1:R0	-	-
IN	Rd, P	In Port	Rd ← P	-	1
OUT	P, Rt	Out Port	P ← Rt	-	1
PUSH	Rt	Push Register on Stack	STACK ← Rt	-	2
POP	Rd	Pop Register from Stack	Rd ← STACK	-	2

*** Μεταφορά μεταξύ Καταχωρητών Εργασίας ή και Καταχωρητή Εργασίας με Θέση μνήμης (δεδομένων και προγράμματος).**

Εντολές αριθμητικών και λογικών πράξεων

ADD	Rd, Rt	Add two Registers	Rd ← Rd + Rt	Z,C,N,V,H	1
ADC	Rd, Rt	Add with Carry two Regs	Rd ← Rd + Rt + C	Z,C,N,V,H	1
ADIW	RdL,K	Add Immediate to Word	RdLh:RdL ← RdhL:RdL + K	Z,C,N,V,S	2
SUB	Rd, Rt	Subtract two Regs	Rd ← Rd - Rt	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Const from Reg	Rd ← Rd - K	Z,C,N,V,H	1
SBC	Rd, Rt	Subtract & C two Regs	Rd ← Rd - Rt - C	Z,C,N,V,H	1
SBCI	Rd, K	Subtract & C Const from Reg	Rd ← Rd - K - C	Z,C,N,V,H	1
SBIW	RdL,K	Subtract Imm from Word	RdhL:RdL ← RdhL:RdL - K	Z,C,N,V,S	2
AND	Rd, Rt	Logical AND Registers	Rd ← Rd ∧ Rt	Z,N,V	1
ANDI	Rd, K	Logical AND Reg & Const	Rd ← Rd ∧ K	Z,N,V	1
OR	Rd, Rt	Logical OR Registers	Rd ← Rd ∨ Rt	Z,N,V	1
ORI	Rd, K	Logical OR Reg & Const	Rd ← Rd ∨ K	Z,N,V	1
EOR	Rd, Rt	Exclusive OR Registers	Rd ← Rd ⊕ Rt	Z,N,V	1
COM	Rd	One's Complement	Rd ← \$FF ⊕ Rd	Z,C,N,V	1
NEG	Rd	Two's Complement	Rd ← \$00 - Rd	Z,C,N,V,H	1
SBR	Rd,K	Set Bit(s) in Register	Rd ← Rd ∨ K	Z,N,V	1
CBR	Rd,K	Clear Bit(s) in Register	Rd← Rd (\$FF - K)	Z,N,V	1
INC	Rd	Increment	Rd ← Rd + 1	Z,N,V	1
DEC	Rd	Decrement	Rd ← Rd - 1	Z,N,V	1
TST	Rd	Test for Zero or Minus	Rd ← Rd ∧ Rd	Z,N,V	1
CLR	Rd	Clear Register	Rd ← Rd ⊕ Rd	Z,N,V	1
SER	Rd	Set Register	Rd ← \$FF	Καμία	1
MUL	Rd, Rt	Multiply Unsigned	R1:R0 ← Rd × Rt	Z,C	2

*** Εκτελούνται μόνο μεταξύ Καταχωρητών Εργασίας ή και Καταχωρητή Εργασίας με σταθερά. Το αποτέλεσμα πάντα πάει σε Καταχωρητή Εργασίας.**

RIJMP	k	Relative Jump	PC ← PC + k + 1	-	2
IJMP		Indirect Jump to (Z)	PC ← Z	-	2
JMP	k	Direct Jump	PC ← k	-	3
RCALL	k	Relative Subroutine Call	PC ← PC + k + 1	-	3
ICALL		Indirect Call to (Z)	PC ← Z	-	3
CALL	k	Direct Subroutine Call	PC ← k	-	4
RET		Subroutine Return	PC ← STACK	-	4
RETI		Interrupt Return	PC ← STACK	I	4

Bit	7	6	5	4	3	2	1	0
Σημεία	I	T	H	S	V	N	Z	C

Καταχωρητής κατάστασης (SREG)

def SYMBOL = VALUE

include "m16def.inc"

Registers: R0-R31

For x=A,B,C,D

I/O direction: DDRx, (1=out,0=in)

PInx = in, PORTx = out|

Bit5 (H): σημαία **δεκαδικού κεραιουμένου** (Half Carry flag). Εμφανίζεται για την ύπαρξη δεκαδικού κεραιουμένου μετά από αριθμητικές πράξεις

Bit4 (S): σημαία **προσήμου** (Sign flag). Εμφανίζεται για το πρόσημο ενός καταχωρητή και ισούται με το XOR των σηματιών αρνητικού προσημου και υπερχείλισης,(S=N xor V)

Οταν έχουμε υπερχείλιση αποτελέσματος που εμφανίζεται >0 είναι στην πραγματικότητα <0

Bit3 (V): σημαία **υπερχείλισης** (Overflow flag) για αριθμητική ουμμητρώματος του δύο

Bit2 (N): σημαία **αρνητικού προσημου** (negative flag)

Bit1 (Z): σημαία **μηδενισμού** (Zero flag). Τίθεται όταν το αποτέλεσμα μιας πράξης είναι 0

Bit0 (C): σημαία **κραιουμένου** (Carry flag).

Εντολές παράκλισης

SBRC	Rt. b	Skip if Bit in Register Cleared	if (Rt(b)=0) *	-	1/ 2/ 3
SBRS	Rt. b	Skip if Bit in Register is Set	if (Rt(b)=1) *	-	1/ 2/ 3
SBIC	P. b	Skip if Bit in I/O Register Cleared	if (P(b)=0) *	-	1/ 2/ 3
SBIS	P. b	Skip if Bit in I/O Register is Set	if (P(b)=1) *	-	1/ 2/ 3

* Σε όλες τις περιπτώσεις που ισχύει η συνθήκη φορτώνεται η νέα τιμή του μετρητή Προγράμματος $PC \leftarrow PC + 2$ or 3.

Ισχύουν για τους Γενικούς Καταχωρητές και για τους Καταχωρητές Θυρών I/O.

Εντολές σύγκρισης

CPSE	Rd,Rt	Compare, Skip if Equal	if (Rd = Rt) $PC \leftarrow PC + 2$ or 3	-	1/2/3
CP	Rd,Rt	Compare	Rd < Rt	Z, N,V,C,H	1
CPC	Rd,Rt	Compare with Carry	Rd < Rt + C	Z, N,V,C,H	1
CPI	Rd,K	Compare Reg with Immediate	Rd < K	Z,N,V,C,H	1

* Μετάξύ Καταχωρητών Εργασίας ή και Καταχωρητή Εργασίας με σταθερά.

Εντολές διακλάδωσης

BRBS	s,k	Branch if Status Flag Set	if (SREG(s) = 1) *	-	1/2
BRBC	s,k	Branch if Status Flag Cleared	if (SREG(s) = 0) *	-	1/2
BREQ	k	Branch if Equal	if (Z = 1) then *	-	1/2
BRNE	k	Branch if Not Equal	if (Z = 0) then *	-	1/2
BRCS	k	Branch if Carry Set	if (C = 1) then *	-	1/2
BRCC	k	Branch if Carry Cleared	if (C = 0) then *	-	1/2
BRSH	k	Branch if Same or Higher	if (C = 0) then *	-	1/2
BRL0	k	Branch if Lower	if (C = 1) then *	-	1/2
BRMI	k	Branch if Minus	if (N = 1) then *	-	1/2
BRPL	k	Branch if Plus	if (N = 0) then *	-	1/2
BRGE	k	Branch if Greater or Equal, Signed	if (N \oplus V= 0) then *	-	1/2
BRLT	k	Branch if Less Than Zero, Signed	if (N \oplus V= 1) then *	-	1/2
BRHS	k	Branch if Half Carry Flag Set	if (H = 1) then *	-	1/2
BRHC	k	Branch if Half Carry Flag Cleared	if (H = 0) then *	-	1/2
BRTS	k	Branch if T Flag Set	if (T = 1) then *	-	1/2
BRTC	k	Branch if T Flag Cleared	if (T = 0) then *	-	1/2
BRVS	k	Branch if Overflow Flag is Set	if (V = 1) then *	-	1/2
BRVC	k	Branch if Overflow Flag is Cleared	if (V = 0) then *	-	1/2
BRIE	k	Branch if Interrupt Enabled	if (I = 1) then *	-	1/2
BRID	k	Branch if Interrupt Disabled	if (I = 0) then *	-	1/2

* Σε όλες τις περιπτώσεις που ισχύει η συνθήκη φορτώνεται η νέα τιμή του μετρητή Προγράμματος $PC \leftarrow PC + k + 1$

Εντολές ελέγχου MCU

NOP	No Operation	-	1
SLEEP	Sleep	Sleep function	1
WDR	Watchdog Reset	WDR/timer function	1
BREAK	Break	For On-chip Debug Only	N/A