Query

#	Opcode	Symbol	Description
0		QRY	Unconditional query
1		FRK	Fork
2		QNZ ARG	Query if argument not zero
3		QNZ *ST	Query if value at ST not zero
4		QNZ *HE	Query if value at HE not zero
5		QNZ *(ST+PO)	Query if value at ST+PO not zero
6		QNZ LI	Query if register is not zero
7		QNZ SQ	Query if register is not zero
8		QNZ RE	Query if register is not zero
9		QNZ TR	Query if register is not zero
10		QNZ CI	Query if register is not zero
11		QNZ PE	Query if register is not zero
12		QNZ HE	Query if register is not zero
13		QNZ ST	Query if register is not zero
14		QNZ EL	Query if register is not zero
15		QNZ PO	Query if register is not zero
16		QZ ARG	Query if argument zero
17		QZ *ST	Query if value at ST zero
18		QZ *HE	Query if value at HE zero
19		QNZ *(ST+PO)	Query if value at ST+PO zero
20		QZ LI	Query if register is zero
21		QZ SQ	Query if register is zero
22		QZ RE	Query if register is zero
23		QZ TR	Query if register is zero
24		QZ CI	Query if register is zero
25		QZ PE	Query if register is zero
26		QZ HE	Query if register is zero
27		QZ ST	Query if register is zero
28		QZ EL	Query if register is zero
29		QZ PO	Query if register is zero
30		QB UB0	If bit is set
31		QB UB1	If bit is set
32		QB UB2	If bit is set
33		QB UB3	If bit is set
34		QB TO0	If bit is set
35		QB TO1	If bit is set
36		QB TO2	If bit is set
37		QB TO3	If bit is set
38		QB CB	If bit is set
39		QB IQ	If bit is set
40		*QB reserved*	*If bit is set*
41		*QB reserved*	*If bit is set*
42		QNB UB0	If bit is cleared
43		QNB UB1	If bit is cleared
44		QNB UB2	If bit is cleared

#
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

45		QNB UB3	If bit is cleared	45
46		QNB TO0	If bit is cleared	46
47		QNB TO1	If bit is cleared	47
48		QNB TO2	If bit is cleared	48
49		QNB TO3	If bit is cleared	49
50		QNB CB	If bit is cleared	50
51		QNB IQ	If bit is cleared	51
52		*QNB reserved*	*If bit is cleared*	52
53		*QNB reserved*	*If bit is cleared*	53
54		QEQ LI	Query if ARG==LI (ALU)	54
55		QGT LI	Query if ARG>LI (ALU)	55
56		QLT LI	Query if ARG<LI (ALU)	56
57		QNN	Query negative number (ALU)	57
58		QFEQ LI	Query if ARG==LI (FPU)	58
59		QFGT LI	Query if ARG>LI (FPU)	59
60		QFLT LI	Query if ARG<LI (FPU)	60
61		QFNAN	Query if ARG is not a number	61
62		QFINF	Query if ARG is infinity	62
63		QIF	Query if secondary branch	63

Operation

Opcode	Symbol	Description		#	Opcode
	ADD	Addition		0	
	ADC	Addition with carry		1	
	SUB	Subtraction		2	
	SBB	Subtraction with borrow		3	
	MUL	ARG = low(A*B)		4	
	MULH	ARG = low(A*B), PE = high(A*B)		5	
	DIV	Integer division		6	
	MOD	Integer modulo		7	
	AND	Bitwise AND	A	8	
	OR	Bitwise OR		9	
	XOR	Bitwise XOR		10	
	NOT	Bitwise NOT		11	
	LAND	Logical AND		12	
	LOR	Logical OR		13	
	LXOR	Logical XOR		14	
	LNOT	Logical NOT	L	15	
	RL	Rotate left (with carry)		16	
	RR	Rotate right		17	
	SL	Shift left (no carry)		18	
	SR	Shift right		19	
	UMIN	Return minimum - unsigned		20	
	UMAX	Return maximum - unsigned		21	
	SMIN	Return minimum - signed		22	
	SMAX	Return maximum - signed		23	
	---	*undefined*	U	24	
	---	*undefined*		25	
	---	*undefined*		26	
	---	*undefined*		27	
	---	*undefined*		28	
	---	*undefined*		29	
	---	*undefined*		30	
	---	*undefined*		31	
	FADD			32	
	FSUB			33	
	FMUL			34	
	FDIV			35	
	FSQR			36	
	FSQRT			37	
	FPOW			38	
	FROOT			39	
	FSIN			40	
	FCOS		F	41	
	FTAN			42	
	FLOG2			43	
	FLOG10			44	

	FLN	
	FABS	
	FTOINT	
	FFROMINT	
	FMIN	
	FMAX	
	FFRAC	
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*
	---	*undefined*

PU

45	
46	
47	
48	
49	
50	
51	
52	
53	
54	
55	
56	
57	
58	
59	
60	
61	
62	
63	
64	
65	
66	
67	
68	
69	
70	
71	
72	
73	
74	
75	
76	
77	
78	
79	
80	
81	
82	
83	
84	
85	
86	
87	
88	
89	
90	
91	
92	
93	
94	

95	
96	
97	
98	
99	
100	
101	
102	
103	
104	
105	
106	
107	
108	
109	
110	
111	
112	
113	
114	
115	
116	
117	
118	
119	
120	
121	
122	
123	
124	
125	
126	
127	

Passtrough action

Symbol	Description
MOV ARG, LI	Register > ARG
MOV ARG, SQ	Register > ARG
MOV ARG, RE	Register > ARG
MOV ARG, TR	Register > ARG
MOV ARG, CI	Register > ARG
MOV ARG, PE	Register > ARG
MOV ARG, HE	Register > ARG
MOV ARG, ST	Register > ARG
MOV ARG, EL	Register > ARG
MOV ARG, PO	Register > ARG
MOV ARG, IC	Register > ARG
MOV ARG, IB	Register > ARG
MOV ARG, SW	Register > ARG
MOV ARG, BS	Register > ARG
MOV ARG, SI	Register > ARG
MOV ARG, SA	Register > ARG
MOV ARG, TC	Register > ARG
MOV ARG, TV0	Register > ARG
MOV ARG, TV1	Register > ARG
MOV ARG, TV2	Register > ARG
MOV ARG, TV3	Register > ARG
MOV ARG, TF0	Register > ARG
MOV ARG, TF1	Register > ARG
MOV ARG, TF2	Register > ARG
MOV ARG, TF3	Register > ARG
MOV LI, ARG	ARG > Register
MOV SQ, ARG	ARG > Register
MOV RE, ARG	ARG > Register
MOV TR, ARG	ARG > Register
MOV CI, ARG	ARG > Register
MOV PE, ARG	ARG > Register
MOV HE, ARG	ARG > Register
MOV ST, ARG	ARG > Register
MOV EL, ARG	ARG > Register
MOV PO, ARG	ARG > Register
MOV IC, ARG	ARG > Register
MOV IB, ARG	ARG > Register
MOV SW, ARG	ARG > Register
MOV BS, ARG	ARG > Register
MOV SI, ARG	ARG > Register
MOV SA, ARG	ARG > Register
MOV TC, ARG	ARG > Register
MOV TV0, ARG	ARG > Register
MOV TV1, ARG	ARG > Register
MOV TV2, ARG	ARG > Register

Retur

#	Opcode
0	RET LI
1	RET SQ
2	RET RE
3	RET TR
4	RET CI
5	RET PE
6	RET HE
7	RET ST
8	RET EL
9	RET PO
10	RET IC
11	RET IB
12	RET SW
13	RET BS
14	RET SI
15	RET SA
16	RET TC
17	RET TV0
18	RET TV1
19	RET TV2
20	RET TV3
21	RET TF0
22	RET TF1
23	RET TF2
24	RET TF3
25	RET *ST
26	RET *HE
27	RET *(ST+LI)
28	RET *(ST+SQ)
29	RET *(ST+RE)
30	RET *(ST+TR)
31	RET *(ST+CI)
32	RET *(ST+PE)
33	RET *(ST+HE)
34	RET *(ST+ST)
35	RET *(ST+EL)
36	RET *(ST+PO)
37	RET *(HE+LI)
38	RET *(HE+SQ)
39	RET *(HE+RE)
40	RET *(HE+TR)
41	RET *(HE+CI)
42	RET *(HE+PE)
43	RET *(HE+HE)
44	RET *(HE+HE)

MOV TV3, ARG	ARG > Register
MOV TF0, ARG	ARG > Register
MOV TF1, ARG	ARG > Register
MOV TF2, ARG	ARG > Register
MOV TF3, ARG	ARG > Register
MOV ARG, *ST	Program+data memory, 32b
MOV *ST, ARG	Program+data memory, 32b
MOV16 ARG, *ST	Program+data memory, 16b
MOV16 *ST, ARG	Program+data memory, 16b
MOV8 ARG, *ST	Program+data memory, 8b
MOV8 *ST, ARG	Program+data memory, 8b
MOV ARG, *HE	IO Interface 16b
MOV *HE, ARG	IO Interface 16b
MOV8 ARG, *HE	IO Interface 8b
MOV8 *HE, ARG	IO inteface 8b
MOV ARG, *(ST+PO)	Program+data memory, 32b
MOV *(ST+PO), ARG	Program+data memory, 32b
MOV16 ARG, *(ST+PO)	Program+data memory, 16b
MOV16 *(ST+PO), ARG	Program+data memory, 16b
MOV8 ARG, *(ST+PO)	Program+data memory, 8b
MOV8 *(ST+PO), ARG	Program+data memory, 8b
MOV ARG, *(HE+PO)	IO Interface 16b
MOV *(HE+PO), ARG	IO Interface 16b
MOV8 ARG, *(HE+PO)	IO Interface 8b
MOV8 *(HE+PO), ARG	IO inteface 8b
PUSH ARG	Push value from register to stack
PUSH SW	Push value from register to stack
PUSH LI	Push value from register to stack
PUSH SQ	Push value from register to stack
PUSH RE	Push value from register to stack
PUSH TR	Push value from register to stack
PUSH CI	Push value from register to stack
PUSH EL	Push value from register to stack
PUSH HE	Push value from register to stack
PUSH ST	Push value from register to stack
PUSH PE	Push value from register to stack
PUSH PO	Push value from register to stack
PUSH *ST	Push value at memory address ST
POP ARG	Pop value from stack to register
POP SW	Pop value from stack to register
POP LI	Pop value from stack to register
POP SQ	Pop value from stack to register
POP RE	Pop value from stack to register
POP TR	Pop value from stack to register
POP CI	Pop value from stack to register
POP EL	Pop value from stack to register
POP HE	Pop value from stack to register
POP ST	Pop value from stack to register
POP EL	Pop value from stack to register
POP PE	Pop value from stack to register

45 RET *(HE+EL)
46 RET *(HE+PO)
47 RET16 *ST
48 RET16 *HE
49 RET16 *(ST+LI)
50 RET16 *(ST+SQ)
51 RET16 *(ST+RE)
52 RET16 *(ST+TR)
53 RET16 *(ST+CI)
54 RET16 *(ST+PE)
55 RET16 *(ST+HE)
56 RET16 *(ST+ST)
57 RET16 *(ST+EL)
58 RET16 *(ST+PO)
59 RET16 *(HE+LI)
60 RET16 *(HE+SQ)
61 RET16 *(HE+RE)
62 RET16 *(HE+TR)
63 RET16 *(HE+CI)
64 RET16 *(HE+PE)
65 RET16 *(HE+HE)
66 RET16 *(HE+HE)
67 RET16 *(HE+EL)
68 RET16 *(HE+PO)
69 RET8 *ST
70 RET8 *HE
71 RET8 *(ST+LI)
72 RET8 *(ST+SQ)
73 RET8 *(ST+RE)
74 RET8 *(ST+TR)
75 RET8 *(ST+CI)
76 RET8 *(ST+PE)
77 RET8 *(ST+HE)
78 RET8 *(ST+ST)
79 RET8 *(ST+EL)
80 RET8 *(ST+PO)
81 RET8 *(HE+LI)
82 RET8 *(HE+SQ)
83 RET8 *(HE+RE)
84 RET8 *(HE+TR)
85 RET8 *(HE+CI)
86 RET8 *(HE+PE)
87 RET8 *(HE+HE)
88 RET8 *(HE+HE)
89 RET8 *(HE+EL)
90 RET8 *(HE+PO)
91
92
93
94

POP *ST	Pop value from stack to memory	95
INC ARG	Increment register	96
INC LI	Increment register	97
INC SQ	Increment register	98
INC HE	Increment register	99
INC ST	Increment register	100
INC PE	Increment register	101
INC PO	Increment register	102
DEC ARG	Decrement register	103
DEC LI	Decrement register	104
DEC SQ	Decrement register	105
DEC HE	Decrement register	106
DEC ST	Decrement register	107
DEC PE	Decrement register	108
DEC PO	Decrement register	109
		110
		111
		112
SETB UB0	Set user bit 0	113
SETB UB1	Set user bit 1	114
SETB UB2	Set user bit 2	115
SETB UB3	Set user bit 3	116
SETB IE	Set bit Interrupt Enable	117
SETB IP	Set bit interrupt Protect	118
SETB CB	Set bit Carry/Borrow	119
CLR UB0	Clear user bit 0	120
CLR UB1	Clear user bit 1	121
CLR UB2	Clear user bit 2	122
CLR UB3	Clear user bit 3	123
CLR IE	Clear Interrupt Enable	124
CLR IP	Clear Interrupt Protect	125
CLR CB	Clear bit Carry/borrow	126
RETI		127
		128

·n

Symbol	Description	Registers	R/W
		SQ - square	UB0…3
		LI - line	TO0…3
		RE - rectangle	TR0…3
		TR - triangle	IQ
		CI - circle	CB
		PE - pentagon	SB
		HE - hexagon	
		ST - star	
		EL - ellipse	
		PO - polygon	
		IC	
		IB	
		SW	
		SI	
		SA	
		TV0...3	
		TF0…3	

bits

user bits
timer overflow
timer run
interrupts queued
carry borrow
sign bit