	c0	c1	c2	c3	c4	c5	c6	c7
00c	[ctrl]	[ctrl2]	[flgs]	JMP [ads]	JSR [ads]	[bra]	[bra2]	MOVQ q#m, [ads]
01c	MOV [reg], [ads]	MOV $m, [ads]	MOV $m(PC), [ads]	MOV (reg), [ads]	MOV $m(reg), [ads]	MOV (reg)+, [ads]	MOV -(reg), [ads]	MOV #m, [ads]
02c	ADD [reg], [ads]	ADD $m, [ads]	ADD $m(PC), [ads]	ADD (reg), [ads]	ADD $m(reg), [ads]	ADD (reg)+, [ads]	ADD -(reg), [ads]	ADD #m, [ads]
03c	ADC [reg], [ads]	ADC $m, [ads]	ADC $m(PC), [ads]	ADC (reg), [ads]	ADC $m(reg), [ads]	ADC (reg)+, [ads]	ADC -(reg), [ads]	ADC #m, [ads]
04c	SUB [reg], [ads]	SUB $m, [ads]	SUB $m(PC), [ads]	SUB (reg), [ads]	SUB $m(reg), [ads]	SUB (reg)+, [ads]	SUB -(reg), [ads]	SUB #m, [ads]
05c	SBB [reg], [ads]	SBB $m, [ads]	SBB $m(PC), [ads]	SBB (reg), [ads]	SBB $m(reg), [ads]	SBB (reg)+, [ads]	SBB -(reg), [ads]	SBB #m, [ads]
06c	AND [reg], [ads]	AND $m, [ads]	AND $m(PC), [ads]	AND (reg), [ads]	AND $m(reg), [ads]	AND (reg)+, [ads]	AND -(reg), [ads]	AND #m, [ads]
07c	OR [reg], [ads]	OR $m, [ads]	OR $m(PC), [ads]	OR (reg), [ads]	OR $m(reg), [ads]	OR (reg)+, [ads]	OR -(reg), [ads]	OR #m, [ads]
10c	XOR [reg], [ads]	XOR $m, [ads]	XOR $m(PC), [ads]	XOR (reg), [ads]	XOR $m(reg), [ads]	XOR (reg)+, [ads]	XOR -(reg), [ads]	XOR #m, [ads]
11c	CMP [reg], [ads]	CMP $m, [ads]	CMP $m(PC), [ads]	CMP (reg), [ads]	CMP $m(reg), [ads]	CMP (reg)+, [ads]	CMP -(reg), [ads]	CMP #m, [ads]
12c	ADDQ q#m, [ads]	ADCQ q#m, [ads]	SUBQ q#m, [ads]	SBBQ q#m, [ads]	ANDQ q#m, [ads]	ORQ q#m, [ads]	XORQ q#m, [ads]	CMPQ q#m, [ads]
13c	NOT [ads]	LSH [ads]	RSH [ads]	INC [ads]	DEC [ads]	SEX [ads]	NOP	NOP

	c0	c1	c2	c3	c4	c5	c6	c7
00c	[ctrl]	[ctrl2]	[flgs]	JMP [ads]	JSR [ads]	[bra]	[bra2]	MOVQ q#m, [ads]
01c	MOV [reg], [ads]	MOV $m, [ads]	MOV $m(PC), [ads]	MOV (reg), [ads]	MOV $m(reg), [ads]	MOV (reg)+, [ads]	MOV -(reg), [ads]	MOV #m, [ads]
02c	ADD [reg], [ads]	ADD $m, [ads]	ADD $m(PC), [ads]	ADD (reg), [ads]	ADD $m(reg), [ads]	ADD (reg)+, [ads]	ADD -(reg), [ads]	ADD #m, [ads]
03c	ADC [reg], [ads]	ADC $m, [ads]	ADC $m(PC), [ads]	ADC (reg), [ads]	ADC $m(reg), [ads]	ADC (reg)+, [ads]	ADC -(reg), [ads]	ADC #m, [ads]
04c	SUB [reg], [ads]	SUB $m, [ads]	SUB $m(PC), [ads]	SUB (reg), [ads]	SUB $m(reg), [ads]	SUB (reg)+, [ads]	SUB -(reg), [ads]	SUB #m, [ads]
05c	SBB [reg], [ads]	SBB $m, [ads]	SBB $m(PC), [ads]	SBB (reg), [ads]	SBB $m(reg), [ads]	SBB (reg)+, [ads]	SBB -(reg), [ads]	SBB #m, [ads]
06c	AND [reg], [ads]	AND $m, [ads]	AND $m(PC), [ads]	AND (reg), [ads]	AND $m(reg), [ads]	AND (reg)+, [ads]	AND -(reg), [ads]	AND #m, [ads]
07c	OR [reg], [ads]	OR $m, [ads]	OR $m(PC), [ads]	OR (reg), [ads]	OR $m(reg), [ads]	OR (reg)+, [ads]	OR -(reg), [ads]	OR #m, [ads]
10c	XOR [reg], [ads]	XOR $m, [ads]	XOR $m(PC), [ads]	XOR (reg), [ads]	XOR $m(reg), [ads]	XOR (reg)+, [ads]	XOR -(reg), [ads]	XOR #m, [ads]
11c	CMP [reg], [ads]	CMP $m, [ads]	CMP $m(PC), [ads]	CMP (reg), [ads]	CMP $m(reg), [ads]	CMP (reg)+, [ads]	CMP -(reg), [ads]	CMP #m, [ads]
12c	ADDQ q#m, [ads]	ADCQ q#m, [ads]	SUBQ q#m, [ads]	SBBQ q#m, [ads]	ANDQ q#m, [ads]	ORQ q#m, [ads]	XORQ q#m, [ads]	CMPQ q#m, [ads]
13c	NOT [ads]	LSH [ads]	RSH [ads]	INC [ads]	DEC [ads]	SEX [ads]	NOP	NOP

	CTRL		
	Mode	ASM	Mnemonic
x0	RESET	RST	RST
x1	HALT	HLT	HLT
x2	CALL	CAL q$m	CAL
x3		NOP	
x4		NOP	
x5		NOP	
x6		NOP	
x7	RETURN SUBROUTINE	RTS	RTS

	CTRL2		
	Mode	ASM	Mnemonic
x0	STORE FLAGS	STF [reg]	STF
x1		NOP	
x2		NOP	
x3		NOP	
x4	BYTE SWAP	SWP [reg]	SWP
x5	EXCHANGE	EXG [reg],[reg]	EXG
x6	LINK	LNK [reg],#m	LNK
x7	UNLINK	ULNK [reg]	ULNK

	FLGS		
	Mode	ASM	Mnemonic
x0	AND IMM. FLAGS	ANF $m	ANF
x1	OR IMM. FLAGS	ORF $m	ORF
x2	XOR IMM. FLAGS	XOF $m	XOF
x3	LOAD FLAGS	LDF [reg]	LDF
x4	AND IMM. STATUS	ANT $m	ANT
x5	OR IMM. STATUS	ORT $m	ORT
x6	XOR IMM. STATUS	XOT $m	XOT
x7	LOAD STATUS	LDT [reg]	LDT

	BRA		
	Mode	Mnemonic	Mnemonic
x0	IF N = 1	BIN $m(PC)	BIN
x1	IF Z = 1	BIZ $m(PC)	BIE
x2	IF V = 1	BIV $m(PC)	BIV
x3	IF C = 1	BIC $m(PC)	BIC
x4	IF N != 1	BNN $m(PC)	BNN
x5	IF Z != 1	BNZ $m(PC)	BNE
x6	IF V != 1	BNV $m(PC)	BNV
x7	IF C != 1	BNC $m(PC)	BNC

	BRA2		
	Mode	ASM	Mnemonic
x0	IF A >= B		BGE
x1	IF A > B		BGT
x2	IF A <= B		BLE
x3	IF A < B		BLT
x4		NOP	
x5		NOP	
x6		NOP	
x7		NOP	

	ADS		
	Mode	ASM	Mnemonic
x0	REG	[reg]	
x1	DIR.	$m	
x2	DIR. PC + OFF.	$m(PC)	
x3	REG IND.	(reg)	
x4	REG + OFF.	$m(reg)	
x5	REG, POST INC.	(reg)+	
x6	SP, PRE DEC.	-(reg)	
x7		NOP	

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x																
1x																
2x																
3x	IRQ8	IRQ9	IRQ10	IRQ11	IRQ12	IRQ13	IRQ14	IRQ15								
4x																
5x	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4	IRQ5	IRQ6	IRQ7								
6x	FAULT0	FAULT1	FAULT2	FAULT3	FAULT4	FAULT5	FAULT6	FAULT7								
7x																
8x																
9x																
Ax																
Bx																
Cx																
Dx																
Ex																
Fx																

D0	CLK
D1	GND
D2	CLK1
D3	CLK2
GND	RESERVED
D4	RESERVED
D5	RESERVED
D6	GND
D7	RESERVED
D8	RESERVED
GND	RESERVED
D9	RESERVED
D10	RESERVED
D11	GND
D12	RESERVED
D13	RESERVED
GND	RESERVED
D14	RESERVED
D15	RESERVED
RESERVED	GND
A0	RESERVED
A1	RESERVED
GND	RESERVED
A2	RESERVED
A3	RESERVED
A4	GND
A5	RESERVED
A6	RESERVED
GND	RESERVED
A7	RESERVED
A8	RESERVED
A9	GND
A10	RESERVED
A11	RESERVED
GND	RESERVED
A12	RESERVED
A13	RESERVED
A14	GND
A15	RESERVED
VCC	VCC
DATA	

~PC_ST	ALU_CVAL
~OP1_ST	GND
~OP2_ST	~PC_EN
~IR_ST	COND_POS
GND	~F_ALUin
~MDR_ST	~F_ST
~MEM_ST	~MAR_ST
~REG_ST	GND
~STAT_ST	RESERVED
RESERVED	RST_MODE
GND	DMA_MODE
~F_DOUT	FLT_MODE
~PC_DOUT	FTCH_MODE
~SUM_DOUT	GND
~RSH_DOUT	RESERVED
~SEX_DOUT	RESERVED
GND	RESERVED
~SWP_DOUT	RESERVED
~WRD_DOUT	RESERVED
~MDR_DOUT	GND
~MEM_DOUT	RESERVED
RESERVED	RESERVED
GND	RESERVED
~PC_AOUT	RESERVED
~MAR_AOUT	RESERVED
RESERVED	GND
COND_N	RESERVED
COND_Z	RESERVED
GND	RESERVED
COND_V	RESERVED
COND_C	RESERVED
RESERVED	GND
ALU_S0	RESERVED
ALU_S1	RESERVED
GND	RESERVED
ALU_S2	RESERVED
ALU_S3	RESERVED
ALU_M	GND
ALU_CSEL	RESERVED
VCC	VCC
CONTROL	

RESERVED	RESERVED	1
RESERVED	GND	
RESERVED	RESERVED	
RESERVED	RESERVED	
GND	RESERVED	5
RESERVED	RESERVED	
RESERVED	RESERVED	
RESERVED	GND	
RESERVED	RESERVED	
RESERVED	RESERVED	10
GND	RESERVED	
RESERVED	RESERVED	
RESERVED	RESERVED	
RESERVED	GND	
RESERVED	RESERVED	15
30 PIN		

D0	CTRL0
D1	CTRL1
D2	GND
D3	CTRL2
D4	CTRL3
D5	CTRL4
GND	CTRL5
D6	CTRL6
D7	CTRL7
~A_DOUT	A0
~MEM_DOUT	GND
~SUM_DOUT	A1
~ROM_DOUT	A2
RESERVED	A3
GND	A4
RESERVED	A5
RESERVED	A6
~A_ST	A7
~MEM_ST	GND
~SUM_ST	RESERVED
~MAR_ST	RESERVED
~OP_ST	RESERVED
GND	RESERVED
~PC_ST	RESERVED
RESERVED	RESERVED
RESERVED	RESERVED
RESERVED	GND
RESERVED	RESERVED
RESERVED	RESERVED
RESERVED	RESERVED
GND	RESERVED
RESERVED	RESERVED
RESERVED	RESERVED
RESERVED	RESERVED
RESERVED	GND
RESERVED	RESERVED
CLK1	RESERVED
CLK2	RESERVED
GND	RESERVED
VCC	VCC