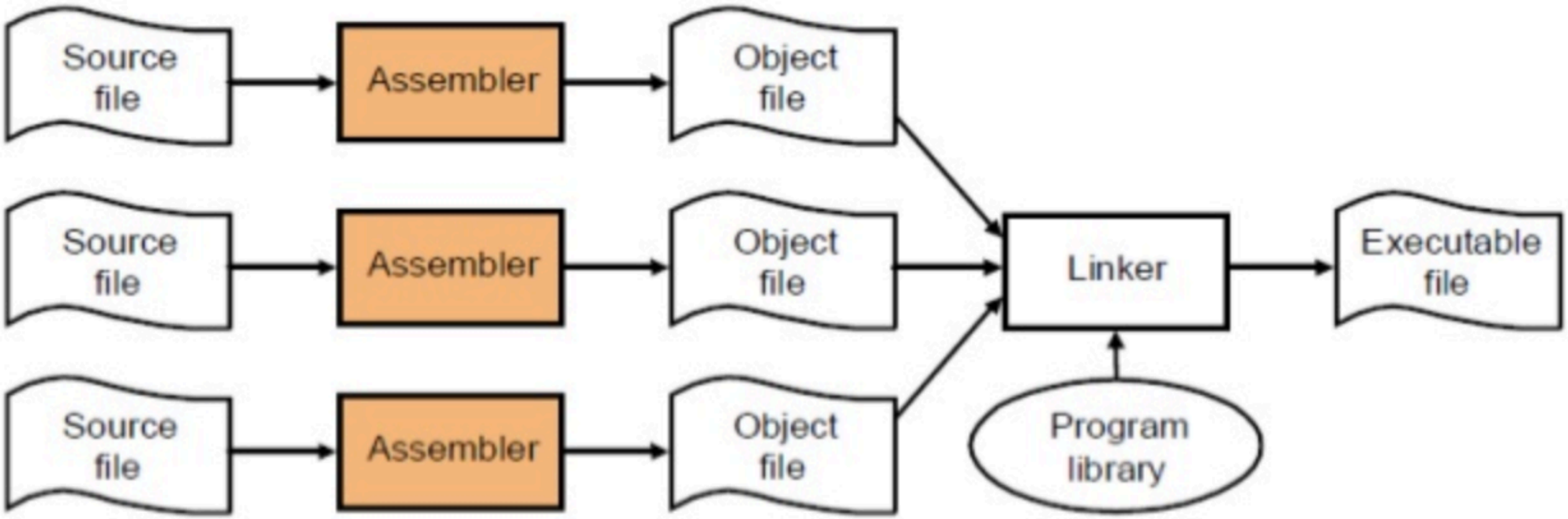
Producing Binary Files



Object Files and Libraries

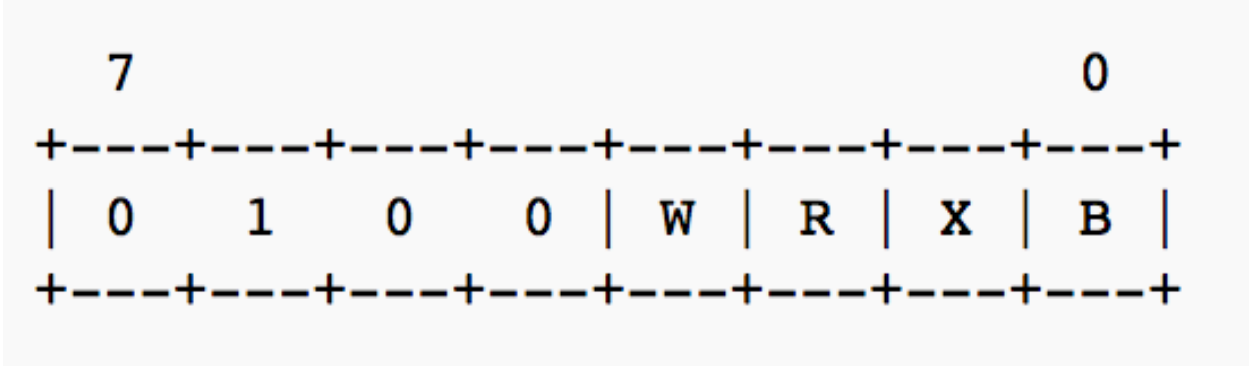
- gcc –S produces assembly
- gcc –c produces object files
- gcc f1.o f2.o links f1 and f2 to make an executable
- nm f1.o lists the ”symbols” of the object f1
- objdump –d f1.o shows the binary representation of f1

Instruction Encoding

	REX								OpCode	ModRM								Immediate
	0	1	0	0	W	R	0	B	(1 or 2 bytes)	Mod (2 bits)		Reg (3 bits)			r/m (3 bits)			
Example (addq):	0	1	0	0	1	0	0	0	83	0	0	0	0	0	0	0	0	0000 1101

MOD	Meaning
00	Register indirect addressing mode or SIB with no displacement (when R/M = 100) or Displacement only addressing mode (when R/M = 101).
01	One-byte signed displacement follows addressing mode byte(s).
10	Four-byte signed displacement follows addressing mode byte(s).
11	Register addressing mode.

REX Prefix



Field	Length	Description
0100	4 bits	Fixed bit pattern
W	1 bit	When 1, a 64-bit operand size is used. Otherwise, when 0, the default operand size is used (which is 32-bit for most but not all instructions).
R	1 bit	This 1-bit value is an extension to the *MODRM.reg* field.
X	1 bit	This 1-bit value is an extension to the *SIB.index* field.
B	1 bit	This 1-bit value is an extension to the *MODRM.rm* field or the *SIB.base* field.

Registers

REG Value	Register if data size is eight bits	Register if data size is 16-bits	Register if data size is 32 bits
000	al	ax	eax
001	cl	cx	ecx
010	dl	dx	edx
011	bl	bx	ebx
100	ah	sp	esp
101	ch	bp	ebp
110	dh	si	esi
111	bh	di	edi

Instruction Encoding

	REX								OpCode	ModRM								Immediate
	0	1	0	0	W	R	0	B	(1 or 2 bytes)	Mod (2 bits)		Reg (3 bits)			r/m (3 bits)			
Example (movq):	0	1	0	0	1	0	0	0	c7	1	1	0	0	0	0	0	0	0001 0101 …

MOD	Meaning
00	Register indirect addressing mode or SIB with no displacement (when R/M = 100) or Displacement only addressing mode (when R/M = 101).
01	One-byte signed displacement follows addressing mode byte(s).
10	Four-byte signed displacement follows addressing mode byte(s).
11	Register addressing mode.

Motorola 68000 CPU Opcodes

Mnemonic	Size	Single Effective Address Operation Word	Data
ORI to CCR	B	0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0	B I
ORI to SR	W	0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0	W I
ORI	B W L	0 0 0 0 0 0 0 0 S M Xn	/ I
ANDI to CCR	B	0 0 0 0 0 0 1 0 0 0 1 1 1 1 1 0 0	B I
ANDI to SR	W	0 0 0 0 0 0 1 0 0 1 1 1 1 1 0 0	W I
ANDI	B W L	0 0 0 0 0 0 1 0 S M Xn	/ I
SUBI	B W L	0 0 0 0 0 1 0 0 S M Xn	/ I
ADDI	B W L	0 0 0 0 0 1 1 0 S M Xn	/ I
EORI to CCR	B	0 0 0 0 1 0 1 0 0 0 1 1 1 1 1 0 0	B I
EORI to SR	W	0 0 0 0 1 0 1 0 0 1 1 1 1 1 0 0	W I
EORI	B W L	0 0 0 0 1 0 1 0 S M Xn	/ I
CMPI	B W L	0 0 0 0 1 1 0 0 S M Xn	/ I
BTST	B L	0 0 0 0 1 0 0 0 0 0 M Xn	B N
BCHG	B L	0 0 0 0 1 0 0 0 0 1 M Xn	B N
BCLR	B L	0 0 0 0 1 0 0 0 1 0 M Xn	B N
BSET	B L	0 0 0 0 1 0 0 0 1 1 M Xn	B N
BTST	B L	0 0 0 0 Dn 1 0 0 M Xn	B N
BCHG	B L	0 0 0 0 Dn 1 0 1 M Xn	B N
BCLR	B L	0 0 0 0 Dn 1 1 0 M Xn	B N
BSET	B L	0 0 0 0 Dn 1 1 1 M Xn	B N
MOVEP	W L	0 0 0 0 Dn 1 D S 0 0 1 An	W D
MOVEA	W L	0 0 S An 0 0 1 M Xn	
MOVE	B W L	0 0 S Xn M M Xn	
MOVE from SR	W	0 1 0 0 0 0 0 0 1 1 M Xn	
MOVE to CCR	B	0 1 0 0 0 1 0 0 1 1 M Xn	
MOVE to SR	W	0 1 0 0 0 1 1 0 1 1 M Xn	
NEGX	B W L	0 1 0 0 0 0 0 0 S M Xn	
CLR	B W L	0 1 0 0 0 0 1 0 S M Xn	
NEG	B W L	0 1 0 0 0 1 0 0 S M Xn	
NOT	B W L	0 1 0 0 0 1 1 0 S M Xn	
EXT	W L	0 1 0 0 1 0 0 0 1 S 0 0 0 Dn	
NBCD	B	0 1 0 0 1 0 0 0 0 0 M Xn	
SWAP	W	0 1 0 0 1 0 0 0 0 1 0 0 0 Dn	
PEA	L	0 1 0 0 1 0 0 0 0 1 M Xn	
ILLEGAL		0 1 0 0 1 0 1 0 1 1 1 1 1 1 0 0	
TAS	B	0 1 0 0 1 0 1 0 1 1 M Xn	
TST	B W L	0 1 0 0 1 0 1 0 S M Xn	
TRAP		0 1 0 0 1 1 1 0 0 1 0 0 Vector	
LINK	W	0 1 0 0 1 1 1 0 0 1 0 1 0 An	W D
UNLK		0 1 0 0 1 1 1 0 0 1 0 1 1 An	
MOVE USP	L	0 1 0 0 1 1 1 0 0 1 1 0 D An	
RESET		0 1 0 0 1 1 1 0 0 1 1 1 0 0 0 0	
NOP		0 1 0 0 1 1 1 0 0 1 1 1 0 0 0 1	
STOP		0 1 0 0 1 1 1 0 0 1 1 1 0 0 1 0	W I

Mnemonic	Size	Single Effective Address Operation Word	Data
RTE		0 1 0 0 1 1 1 0 0 1 1 1 0 0 1 1	
RTS		0 1 0 0 1 1 1 0 0 1 1 1 0 1 0 1	
TRAPV		0 1 0 0 1 1 1 0 0 1 1 1 0 1 1 0	
RTR		0 1 0 0 1 1 1 0 0 1 1 1 0 1 1 1	
JSR		0 1 0 0 1 1 1 0 1 0 M Xn	
JMP		0 1 0 0 1 1 1 0 1 1 M Xn	
MOVEM	W L	0 1 0 0 1 D 0 0 1 S M Xn	W M
LEA	L	0 1 0 0 An 1 1 1 M Xn	
CHK	W	0 1 0 0 Dn 1 1 0 M Xn	
ADDQ	B W L	0 1 0 1 Data 0 S M Xn	
SUBQ	B W L	0 1 0 1 Data 1 S M Xn	
Scc	B	0 1 0 1 Condition 1 1 M Xn	
DBcc	W	0 1 0 1 Condition 1 1 0 0 1 Dn	W D
BRA	B W	0 1 1 0 0 0 0 0 Displacement	W D
BSR	B W	0 1 1 0 0 0 0 1 Displacement	W D
Bcc	B W	0 1 1 0 Condition Displacement	W D
MOVEQ	L	0 1 1 1 Dn 0 Data	
DIVU	W	1 0 0 0 Dn 0 1 1 M Xn	
DIVS	W	1 0 0 0 Dn 1 1 1 M Xn	
SBCD	B	1 0 0 0 Xn 1 0 0 0 0 M Xn	
OR	B W L	1 0 0 0 Dn D S M Xn	
SUB	B W L	1 0 0 1 Dn D S M Xn	
SUBX	B W L	1 0 0 1 Xn 1 S 0 0 M Xn	
SUBA	W L	1 0 0 1 An S 1 1 M Xn	
EOR	B W L	1 0 1 1 Dn 1 S M Xn	
CMPM	B W L	1 0 1 1 An 1 S 0 0 1 An	
CMP	B W L	1 0 1 1 Dn 0 S M Xn	
CMPA	W L	1 0 1 1 An S 1 1 M Xn	
MULU	W	1 1 0 0 Dn 0 1 1 M Xn	
MULS	W	1 1 0 0 Dn 1 1 1 M Xn	
ABCD	B	1 1 0 0 Xn 1 0 0 0 0 M Xn	
EXG	L	1 1 0 0 Xn 1 M 0 0 M Xn	
AND	B W L	1 1 0 0 Dn D S M Xn	
ADD	B W L	1 1 0 1 Dn D S M Xn	
ADDX	B W L	1 1 0 1 Xn 1 S 0 0 M Xn	
ADDA	W L	1 1 0 1 An S 1 1 M Xn	
ASd	B W L	1 1 1 0 0 0 0 D 1 1 M Xn	
LSd	B W L	1 1 1 0 0 0 1 D 1 1 M Xn	
ROXd	B W L	1 1 1 0 0 1 0 D 1 1 M Xn	
ROd	B W L	1 1 1 0 0 1 1 D 1 1 M Xn	
ASd	B W L	1 1 1 0 Rotation D S M 0 0 Dn	
LSd	B W L	1 1 1 0 Rotation D S M 0 1 Dn	
ROXd	B W L	1 1 1 0 Rotation D S M 1 0 Dn	
ROd	B W L	1 1 1 0 Rotation D S M 1 1 Dn	

Addressing Mode	Format	M	Xn
Data register	Dn	0 0 0	reg
Address register	An	0 0 1	reg
Address	(An)	0 1 0	reg
Address with Postincrement	(An)+	0 1 1	reg
Address with Predecrement	-(An)	1 0 0	reg
Address with Displacement	(d_{16}, An)	1 0 1	reg
Address with Index	(d_8, An, Xn)	1 1 0	reg
Program Counter with Displacement	(d_{16}, PC)	1 1 1	0 1 0
Program Counter with Index	(d_8, PC, Xn)	1 1 1	0 1 1
Absolute Short	(xxx).W	1 1 1	0 0 0
Absolute Long	(xxx).L	1 1 1	0 0 1
Immediate	#imm	1 1 1	1 0 0

Operation Size	Suffix	S	S	S
Byte	.b	0 0	/	0 1
Word	.w	0 1	0	1 1
Long	.l	1 0	1	1 0

Direction	d	D
Right	R	0
Left	L	1

Rotation	M
Immediate	0
Register	1

Mode	M
Dn	0
-(An)	1

Condition	Mnemonic	Cond
True	T	0 0 0 0
False	F	0 0 0 1
Higher	HI	0 0 1 0
Lower or Same	LS	0 0 1 1
Carry Clear	CC	0 1 0 0
Carry Set	CS	0 1 0 1
Not Equal	NE	0 1 1 0
Equal	EQ	0 1 1 1
Overflow Clear	VC	1 0 0 0
Overflow Set	VS	1 0 0 1
Plus	PL	1 0 1 0
Minus	MI	1 0 1 1
Greater or Equal	GE	1 1 0 0
Less Than	LT	1 1 0 1
Greater Than	GT	1 1 1 0
Less or Equal	LE	1 1 1 1

Data Type	Letter
Immediate	I
Bit Index	N
Displacement	D
Optional Displacement	D
Register List Mask	M

Data Size	Letter
Byte	B
Word	W
Long	L
Any	/

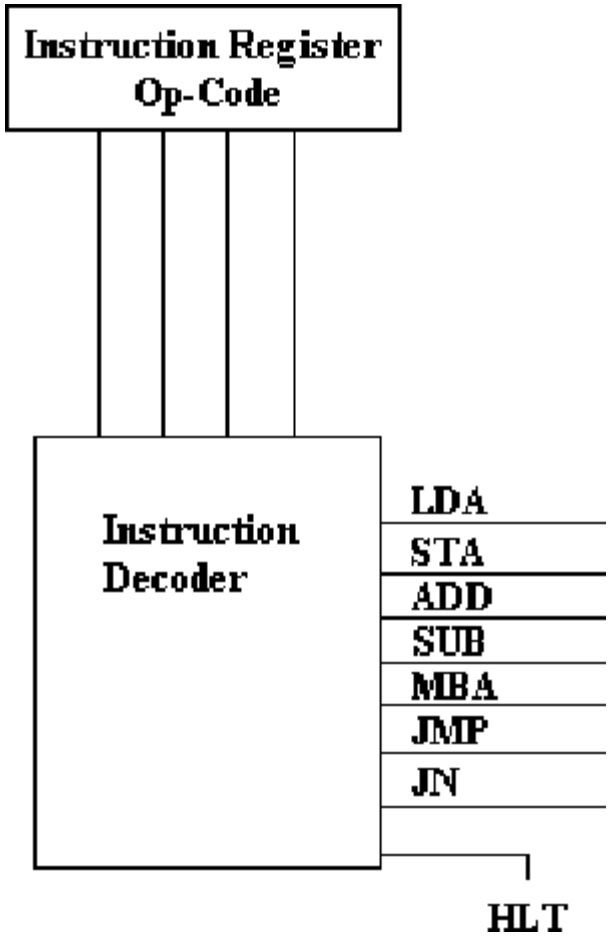
Direction	D	D
Register to memory	0	1
Memory to register	1	0

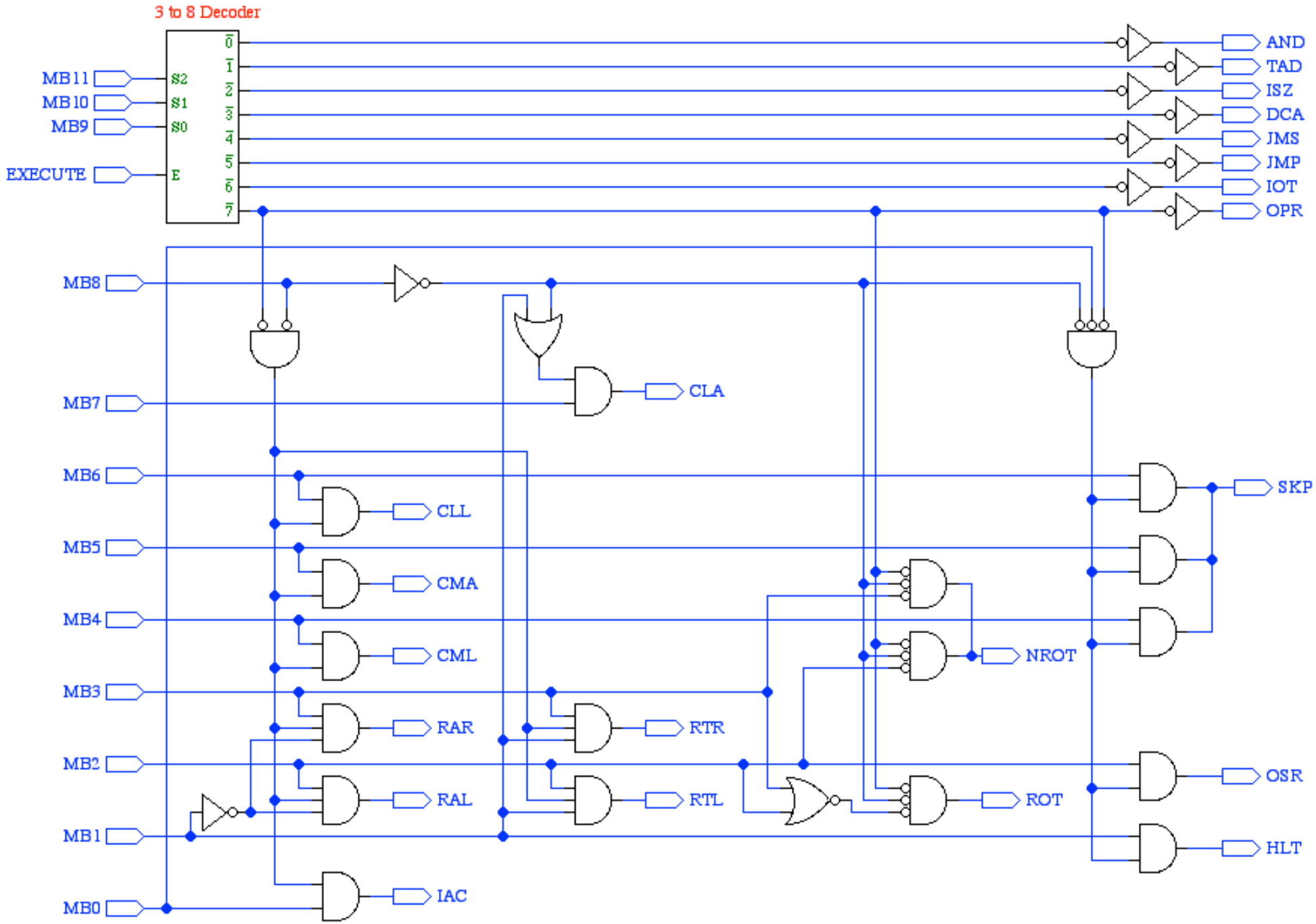
Direction	D
Dn ♦ <ea> → Dn	0
<ea> ♦ Dn → <ea>	1

Mode	Register List Mask															
Postincrement	A7	A6	A5	A4	A3	A2	A1	A0	D7	D6	D5	D4	D3	D2	D1	D0
Predecrement	D0	D1	D2	D3	D4	D5	D6	D7	A0	A1	A2	A3	A4	A5	A6	A7

Brief Extension Word					
M	Xn	S	0	0	0	Displacement
---	---	---	---	---	---	---

Instruction Decoding




3 to 8 Decoder
MB11
MB10
MB9
EXECUTE
S2
S1
S0
E
0
1
2
3
4
5
6
7
AND
TAD
ISZ
DCA
JMS
JMP
IOT
OPR
MB8
MB7
MB6
MB5
MB4
MB3
MB2
MB1
MB0
CLA
CLL
CMA
CML
RAR
RAL
IAC
RTR
RTL
NROT
ROT
SKP
OSR
HLT