ECE 3120
Computer Systems
Instructions

Manjeera Jeedigunta

http://blogs.cae.tntech.edu/msjeedigun21

Email: msjeedigun21@tntech.edu

Tel: 931-372-6181, Prescott Hall 120

- Prev:
  - 68HCS12 Addressing Modes
- Today:
  - 68HCS12 instructions

The LOAD and STORE Instructions

- The LOAD instruction copies the contents of a memory location or places an immediate value into an accumulator or a CPU register.
- STORE instructions save the contents of a CPU register into a memory location.
- N and Z flags of the CCR register are automatically updated and the V flag is cleared.
- All except for the relative mode can be used to select the memory location or value to be loaded into an accumulator or CPU register.
- All except for the relative and immediate modes can be used to select memory location to store contents of the CPU register. For example,

```
ldaa  0,X
staa  $20
stx  $8000
ldd  #100
```

Table 1.4 Load and store instructions

Mnemonic	Function	Operation
LDAA	Load A	(M) ⇒ A
LDAB	Load B	(M) ⇒ B
LDD	Load D	(M:M+1) ⇒ (A:B)
LDS	Load SP	(M:M+1) ⇒ SP
LDX	Load index register X	(M:M+1) ⇒ X
LDY	Load index register Y	(M:M+1) ⇒ X
LEAS	Load effective address into SP	Effective address ⇒ SP
LEAX	Load effective address into X	Effective address ⇒ X
LEAY	Load efective address into Y	Effective address ⇒ Y

Store Instructions

Mnemonic	Function	Operation
STAA	Store A	(A) ⇒ M
STAB	Store B	(B) ⇒ M
STD	Store D	(A) ⇒ M, (B) ⇒ M+1
STS	Store SP	(SP) ⇒ M, M+1
STX	Store X	(X) ⇒ M:M+1
STY	Store Y	(Y) ⇒ M:M+1

Transfer and Exchange Instructions

- **Transfer instructions copy the contents of a CPU register or accumulator into another CPU register or accumulator.**
- TFR **is the universal transfer instruction, but other mnemonics are accepted for compatibility with the 68HC11.**
- **The TAB and TBA instructions affect the N, Z, and V condition code bits.**
- **The** TFR **instruction does not affect any condition code bits. For example,**

```
TFR  D,X                    ; [D] ⇒ X
TFR  A,B                    ; [A] ⇒ B
```

- **The** EXG **instruction exchanges the contents of a pair of registers or accumulators. For example,**

```
exg  A, B
exg  D,X
```

- **The** SEX **instruction sign-extend an 8-bit two's complement number into a 16-bit number so that it can be used in 16-bit signed operations. For example,**

```
SEX A,X
```

Move Instructions

- These instructions move data bytes or words from a source to a destination in memory.

- Six combinations of immediate, extended, and index addressing modes are allowed to specify the source and destination addresses:

IMM ⇒ EXT,	**IMM ⇒ IDX,**	**EXT ⇒ EXT,**
EXT ⇒ IDX,	**IDX ⇒ EXT,**	**IDX ⇒ IDX**

- Examples:

```
movb    $100,$800
movw    0,X, 0,Y
```

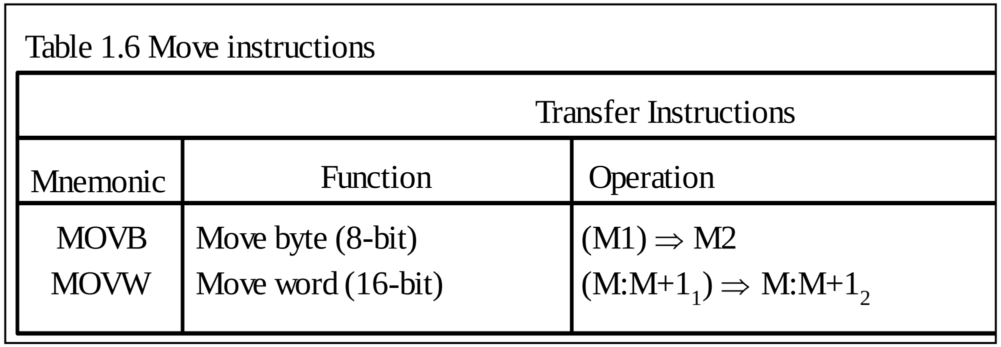

Table 1.6 Move instructions

Transfer Instructions		
Mnemonic	Function	Operation
MOVB MOVW	Move byte (8-bit) Move word (16-bit)	(M1) ⇒ M2 $(M:M+1_1) \Rightarrow M:M+1_2$

Add and Subtract Instructions

- **These instructions perform fundamental arithmetic operations.**

- **The destinations of these instructions are always a CPU register or accumulator.**

- **For example,**

```
adda $800          ; A ⇐ [A] + [$800]
adca $800          ; A ⇐ [A] + [$800] + C
suba $802          ; A ⇐ [A] + [$802]
sbca $800          ; A ⇐ [A] - [$800] - C
```

Table 1.7 Add and subtract instructions

Add Instructions		
Mnemonic	Function	Operation
ABA	Add B to A	(A) + (B) ⇒ A
ABX	Add B to X	(B) + (X) ⇒ X
ABY	Add B to Y	(B) + (Y) ⇒ Y
ADCA	Add with carry to A	(A) + (M) + C ⇒ A
ADCB	Add with carry to B	(B) + (M) + C ⇒ B
ADDA	Add without carry to A	(A) + (M) ⇒ A
ADDB	Add without carry to B	(B) + (M) ⇒ B
ADDD	Add without carry to D	(A:B) + (M:M+1) ⇒ A:B
Subtract Instructions		
Mnemonic	Function	Operation
SBA	Subtract B from A	(A) - (B) ⇒ A
SBCA	Subtract with borrow from A	(A) - (M) - C ⇒ A
SBCB	Subtract with borrow from B	(B) - (M) - C ⇒ B
SUBA	Subtract memory from A	(A) - (M) ⇒ A
SUBB	Subtract memory from B	(B) - (M) ⇒ B
SUBD	Subtract memory from D	(D) - (M:M+1) ⇒ D