

4	4.10 A	Simple I	Program	CSC3501 - 5	S.J. Park
	Consider	the simple		niu on holouu	۱۸/
			MARIE program of monic instructions		
		0 - 106 (he		sioreu al a	uui
e	5355 10	0 - 100 (ne.	<b>~</b> ).		
ĺ	Address	Instruction	Binary Contents of Memory Address	Hex Contents of Memory	
	Address	Instruction			
			Memory Address	of Memory	
0	100	Load 104	Memory Address 0001000100000100	of Memory 1104	
0	100 101	Load 104 Add 105	Memory Address 0001000100000100 0011000100000101	of Memory 1104 3105	
	100 101 102	Load 104 Add 105 Store 106	Memory Address 0001000100000100 0011000100000101 0100000100000110	of Memory 1104 3105 4106	
þ	100 101 102 103	Load 104 Add 105 Store 106 Halt	Memory Address 0001000100000100 0011000100000101 0100000100000110 0111000000	of Memory 1104 3105 4106 7000	

## 4.10 A Simple Program

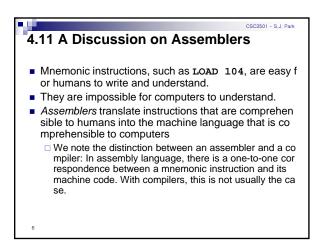
• Let's look at what happens inside the computer when our program runs.

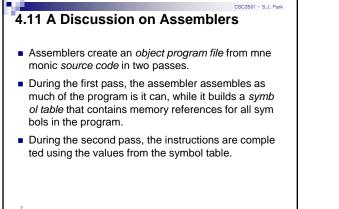
CSC3501 - S.J. Park

This is the LOAD 104 instruction:

Step	RTN	PC	IR	MAR	MBR	AC
(initial values)		100				
Fetch	MAR - PC	100		100	******	
	IR ← M[MAR]	100	1104	100		****
	PC ← PC + 1	101	1104	100		*****
Decode	MAR ← IR [11-0]	101	1104	104		
	(Decode IR[15-12])	101	1104	104	******	
Get operand	MBR ← M[MAR]	101	1104	104	0023	*****
Execute	AC - MBR	101	1104	104	0023	0023

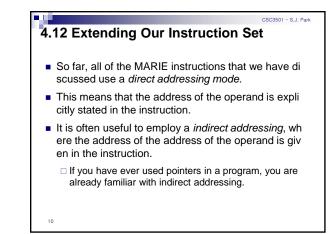
Our accord	l instruction is A		105.			
Our second	I INSTITUCTION IS A	עע	105.			
Step	RTN	PC	IR	MAR	MBR	AC
(initial values)		101	1104	104	0023	0023
Fetch	MAR - PC	101	1104	101	0023	0023
	IR ← M[MAR]	101	3105	101	0023	0023
	PC ← PC + 1	102	3105	101	0023	0023
Decode	MAR ← IR[11-0]	102	3105	105	0023	0023
	(Decode IR[15-12])	102	3105	105	0023	0023
Get operand	MBR	102	3105	105	FFE9	0023
Execute	AC ← AC + MBR	102	3105	105	FFE9	000C

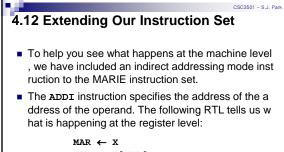




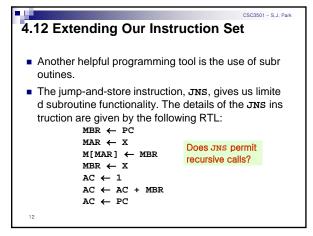
<ul> <li>Consider our example progr</li> </ul>	Address	Instruc	ction
am (top).	100	Load	х
Note that we have included tw o directives HEX and DEC that	101	Add	Y
specify the radix of the consta	102	Store	Z
nts.	103	Halt	
- During the first pass, we have	104 X,	DEC	35
<ul> <li>During the first pass, we hav e a symbol table and the par</li> </ul>	105 Y,	DEC	-23
tial instructions shown at the	106 Z,	HEX	000
bottom.	1 X		
Y 105	3 Y		
	2 Z		
Z 106	7000		

4.11 A Discu	ssion on A	Assembl	CSC3501 - S.J. Park
<ul> <li>After the second</li> </ul>		Address	Instruction
assembly is co	mplete.	100	Load X
		101	Add Y
1104		102	Store Z
3105		103	Halt
		104 X,	DEC 35
2106		105 Y,	DEC -23
7000		106 Z,	HEX 0000
0023	X 104	1	
FFE 9	Y 105	1	
0000	Z 106	1	
9		-	





```
MBR \leftarrow M[MAR]MAR \leftarrow MBRMBR \leftarrow M[MAR]AC \leftarrow AC + MBR
```



## CSC3501 - S.J. Park 4.12 Extending Our Instruction Set Our last helpful instruction is the CLEAR instruction. All it does is set the contents of the accumulator to a ll zeroes. This is the RTL for CLEAR: AC ← 0

• We put our new instructions to work in the program on the following slide.

13

			_	_	CSC3501 - S.J. Park	
4.12 Extending Our Instruction Set						
		•				
100		LOAD Addr	10E	1	SKIPCOND 000	
101		STORE Next	10F	1	JUMP Loop	
102		LOAD Num	110	1	HALT	
103		SUBT One	111	Addr	HEX 118	
104	1	STORE Ctr	112	Next	HEX 0	
105	Loop	LOAD Sum	113	Num	DEC 5	
106	Í	ADDI Next	114	Sum	DEC 0	
107	Í	STORE Sum	115	Ctr	HEX 0	
108	Í	LOAD Next	116	One	DEC 1	
109	i	ADD One	117	i	DEC 10	
10A	i	STORE Next	118	i	DEC 15	
10B	i	LOAD Ctr	119	İ	DEC 2	
10C	i i	SUBT One	11A	i i	DEC 25	
10D	i i	STORE Ctr	11B	i i	DEC 30	
14						
1.44						