Microprogram Design

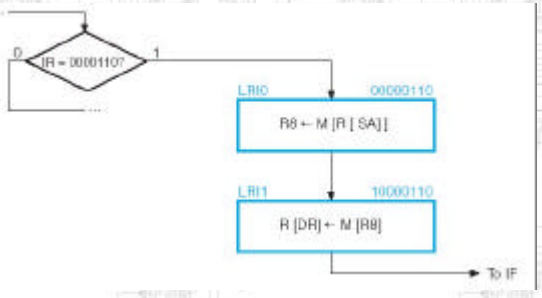
- Use ASM from Lecture 15 to design the Microprograms
- See Symbolic/Binary Microprogram on next slide

Symbolic/Binary Microprogram

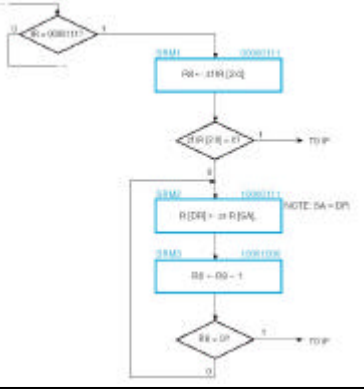
Address	NXT ADD	MS	MC	IL	PI	PL	TD	TA	TB	MB	FS	MD	RW	MM	MW
IF	EX0	CNT	—	LD0	INP	NLP	—	—	—	—	—	—	NW	PC	NW
EX0	—	NXT	OPC	NLI	NLP	NLP	—	—	—	—	—	—	NW	—	NW
ADX	IF	NXT	NXA	NLI	NLP	NLP	DR	SA	—	—	Constant	F = A + B	FnRt	WR	—	NW
LD	IF	NXT	NXA	NLI	NLP	NLP	DR	SA	—	—	—	—	Data	WR	MA	NW
ST	IF	NXT	NXA	NLI	NLP	NLP	—	—	SA	SB	Register	—	—	NW	MA	WR
INC	IF	NXT	NXA	NLI	NLP	NLP	DR	SA	—	—	—	F = A + 1	FnRt	WR	—	NW
NOT	IF	NXT	NXA	NLI	NLP	NLP	DR	SA	—	—	—	F = Ā	FnRt	WR	—	NW
ADD	IF	NXT	NXA	NLI	NLP	NLP	DR	SA	—	SB	Register	F = A + B	FnRt	WR	—	NW

Address	NXT ADD	MS	MC	IL	PI	PL	TD	TA	TB	MB	FS	MD	RW	MM	MW
192	193	000	0	1	1	0	0	0	0	0	00000	0	0	1	0
193	000	001	1	0	0	0	0	0	0	0	00000	0	0	0	0
000	192	001	0	0	0	0	0	0	0	1	00000	0	1	0	0
001	192	001	0	0	0	0	0	0	0	0	00000	1	1	0	0
002	192	001	0	0	0	0	0	0	0	0	00000	0	0	0	1
003	192	001	0	0	0	0	0	0	0	0	00001	0	1	0	0
004	192	001	0	0	0	0	0	0	0	0	01110	0	1	0	0
005	192	001	0	0	0	0	0	0	0	0	00010	0	1	0	0

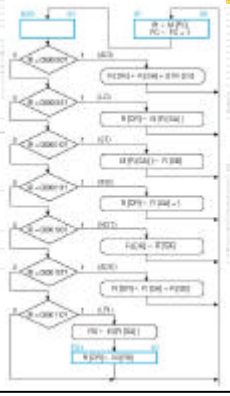
Indirect Instruction ASM



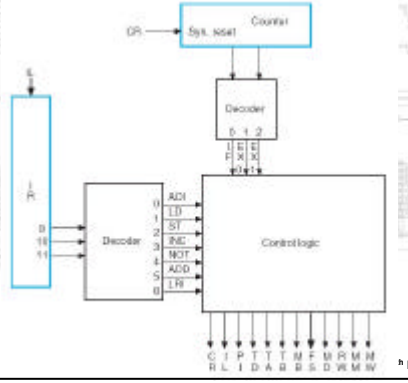
Right-Shift Instruction ASM



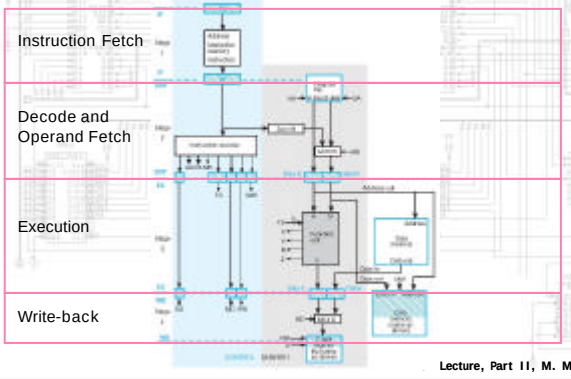
Hardwired Multiple-Cycle Control



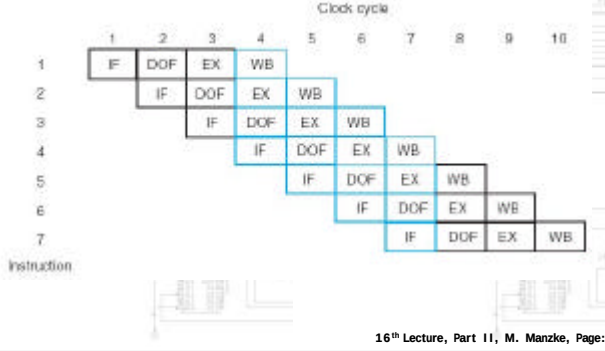
Hardwired Control Unit



Pipelined (based on single-cycle)



Pipelined Execution Pattern



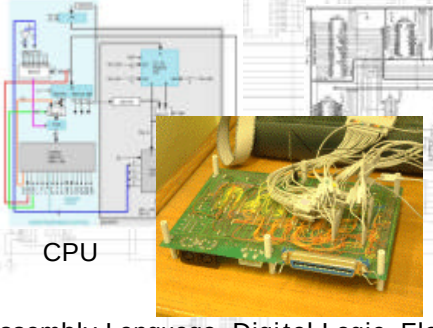
Project 2

- Implement the following instructions:
  - ADI
  - LD
  - ST
  - INC
  - NOT
  - ADD
- Implement the instructions from Problem:
  - 8-40 unconditional jump (Mano & Kime)
  - 8-41 conditional branch (Mano & Kime)

Exam

- Work through the problems at the end of chapter 8

Computer Architecture and Microprocessor Systems



CPU

Board Level

Assembly Language, Digital Logic, Electrotechnology