CS430 Computer Architecture

Spring 2015

Chapter 2
Computer Evolution
and
Performance

- Reading: pp. 16-49
- Good Problems to Work: 2.1, 2.3, 2.4, 2.8, 2.9

Brief History

- ENIAC (Electronic Numerical Integrator and Computer)
  - Designed by: John Mauchly and John Presper Eckert
  - Reason: Response to wartime needs
  - Completed: 1946
  - Specifics: 30 tons, 15,000 sq ft, 18000 vacuum tubes, 5000 additions/sec decimal machine with 20 accumulators each accumulator could hold a 10-digit decimal number

ENIAC Programming

- ENIAC involved plugging and unplugging cables.
- John von Neumann "If however, the program could be stored somehow and held in memory along with the data, this process would not be necessary. " known as the stored-program concept

IAS Computer

Designed by: John von Neumann

Reason: Incorporate the stored-program concept

Completed: 1952

Specifics: 1000 words (40-bit words) of storage, both data and instructions are stored, binary representations

IAS Computer

- von Neumann helped engineer the IAS (completed in 1952) utilizing the stored-program concept

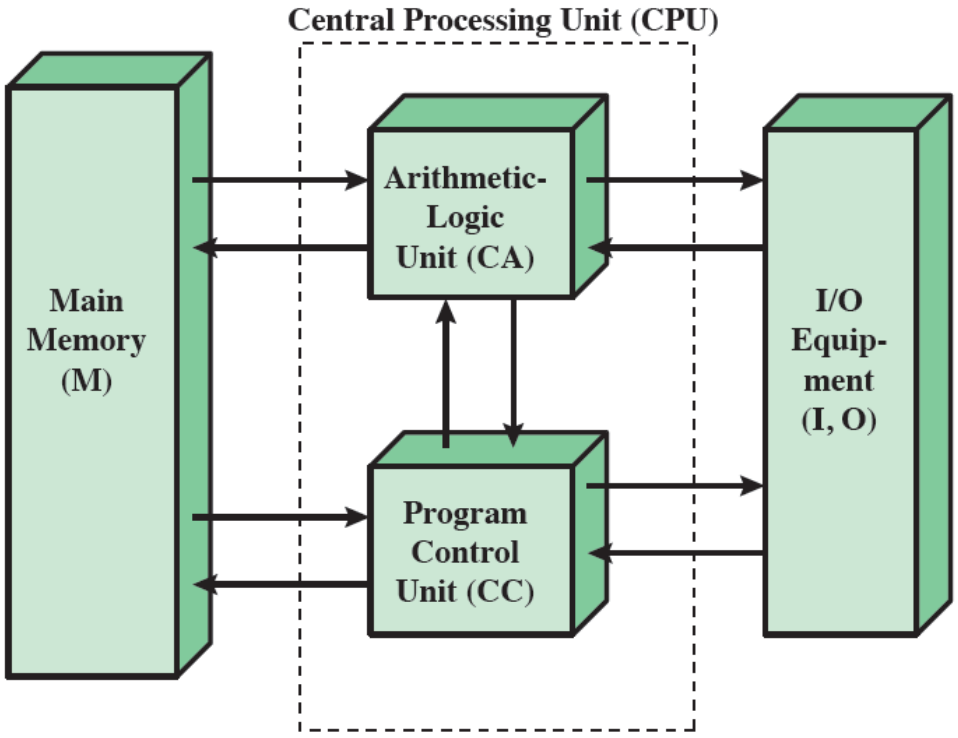


Figure 2.1 Structure of the IAS Computer

IAS Computer

- Main memory - stores both data and instructions
- Arithmetic and logic unit (ALU) - operates on binary data
- Control unit (CU) - interprets the instruction in memory and causes the instruction to be executed
- Input/Output (I/O) - operated by CU

IAS Memory

- A word (40-bits) of storage can represent either an instruction or a number.

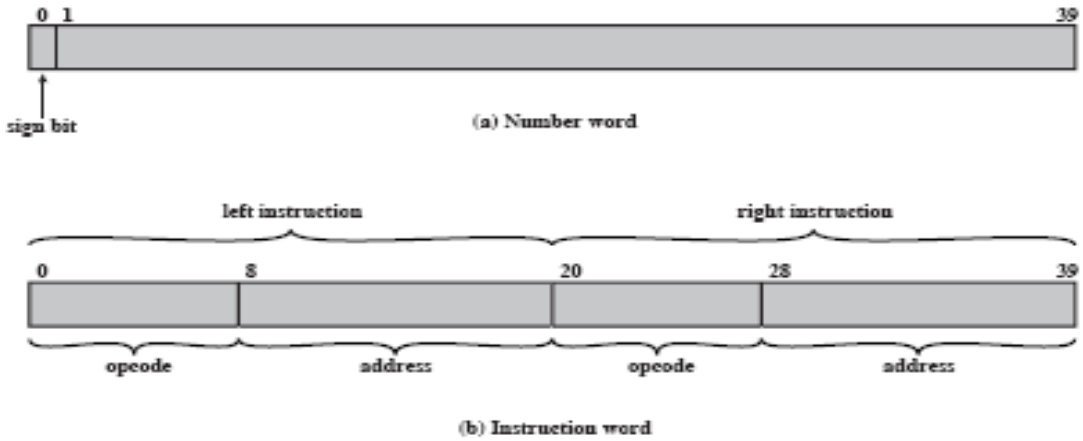


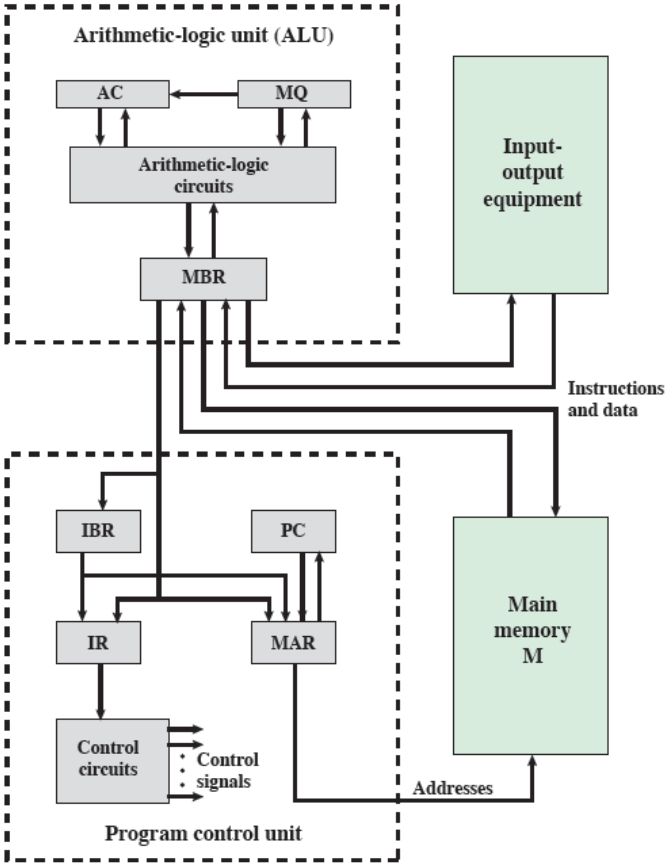
Figure 2.2 IAS Memory Formats

- An instruction word really contains two instructions (left and right instructions)

Special IAS Registers

- Memory Buffer Register (MBR) - used to store/retrieve a word to/from memory
- Memory Address Register (MAR) - specifies the address used to store/retrieve a word to/from MBR
- Instruction Register (IR) - contains opcode (8-bit) of the current instruction being executed
- Instruction Buffer Register (IBR) - used to hold right instruction
- Program Counter (PC) - contains address of next instruction to be executed
- Accumulator (AC) and Multipier Quotient (MQ) - holds results of ALU operations. For a multiply of 2 words, the AC holds the most significant 40-bits and the MQ holds the least significant 40-bits.

Expanded IAS Computer Structure



Instruction Cycle

1. Fetch the next instruction
2. Execute the next instruction
3. go to 1.

Fetch Cycle

1. Load the opcode portion of the next instruction into the IR
2. Load the address portion of the next instruction into the MAR

Note: The next instruction can come from the IR, IBR, or MBR (to the IR, MAR & IBR)

Execute Cycle

1. Control circuitry decodes the opcode
2. Control signals are sent based on the decoded opcode

Fetch/Execute Partial Flowchart

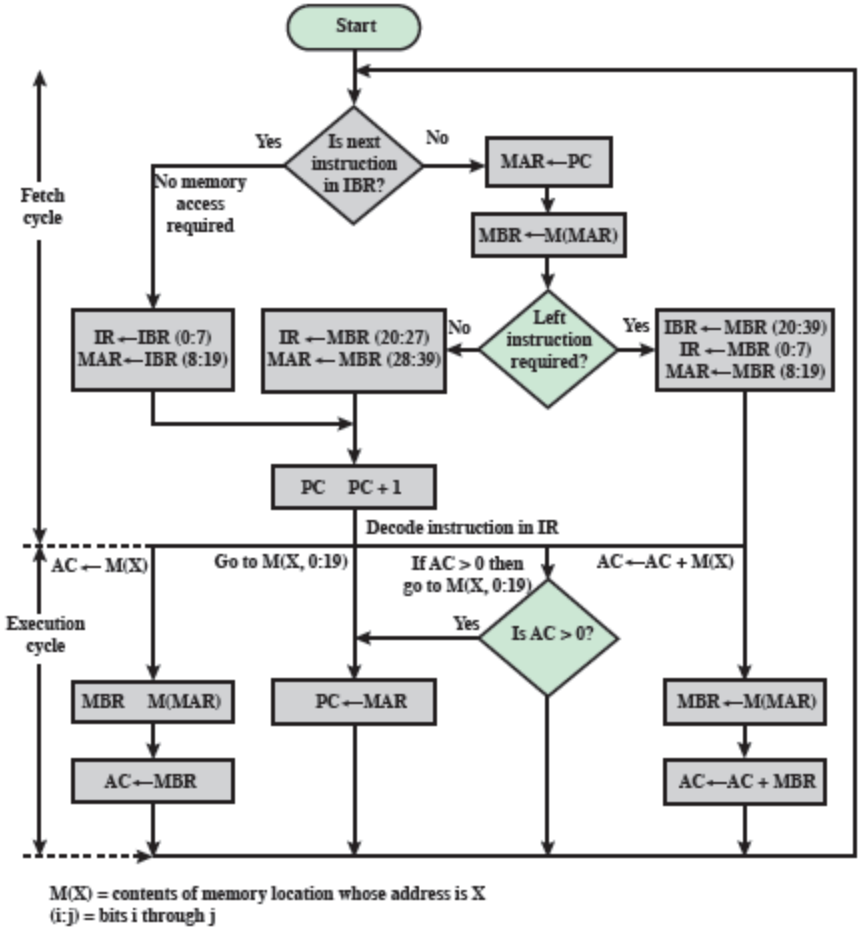


Figure 2.4 Partial Flowchart of IAS Operation

IAS Instructions

- IAS Instructions are grouped as:
  1. data transfer - move data between MEM & ALU or two ALU registers
  2. unconditional branch - branch to some location unconditionally. Execution may not be sequential in this case
  3. conditional branch - branch is made based on some condition (allows decisions to be made)
  4. arithmetic - ALU operations
  5. address modify - addresses in an instruction can be modified

IAS Instruction Set

Table 2.1 The IAS Instruction Set

Instruction Type	Opcode	Symbolic Representation	Description
Data transfer	00001010	LOAD MQ	Transfer contents of register MQ to the accumulator AC
	00001001	LOAD MQ,M(X)	Transfer contents of memory location X to MQ
	00100001	STOR M(X)	Transfer contents of accumulator to memory location X
	00000001	LOAD M(X)	Transfer M(X) to the accumulator
	00000010	LOAD –M(X)	Transfer –M(X) to the accumulator
	00000011	LOAD |M(X)|	Transfer absolute value of M(X) to the accumulator
	00000100	LOAD –|M(X)|	Transfer –|M(X)| to the accumulator
Unconditional branch	00001101	JUMP M(X,0:19)	Take next instruction from left half of M(X)
	00001110	JUMP M(X,20:39)	Take next instruction from right half of M(X)
Conditional branch	00001111	JUMP+ M(X,0:19)	If number in the accumulator is nonnegative, take next instruction from left half of M(X)
	00010000	JUMP+ M(X,20:39)	If number in the accumulator is nonnegative, take next instruction from right half of M(X)
Arithmetic	00000101	ADD M(X)	Add M(X) to AC; put the result in AC
	00000111	ADD |M(X)|	Add |M(X)| to AC; put the result in AC
	00000110	SUB M(X)	Subtract M(X) from AC; put the result in AC
	00001000	SUB |M(X)|	Subtract |M(X)| from AC; put the remainder in AC
	00001011	MUL M(X)	Multiply M(X) by MQ; put most significant bits of result in AC, put least significant bits in MQ
	00001100	DIV M(X)	Divide AC by M(X); put the quotient in MQ and the remainder in AC
	00010100	LSH	Multiply accumulator by 2, i.e., shift left one bit position
	00010101	RSH	Divide accumulator by 2, i.e., shift right one position
Address modify	00010010	STOR M(X,8:19)	Replace left address field at M(X) by 12 rightmost bits of AC
	00010011	STOR M(X,28:39)	Replace right address field at M(X) by 12 rightmost bits of AC

IAS Problem

1. Write the IAS assembly language program that will add the numbers stored at location 100 and 101. Place the result at location 102.
2. Assuming your assembly language program is stored beginning at location 110, fill in the table below.

Instruction	Opcode	Description

IAS Problem

1. Using Figure 2.4 Partial Flowchart of the IAS Operation, what does the instruction cycle for adding two numbers defined above look like?

Computer Generations

Table 2.2 Computer Generations

Generation	Approximate Dates	Technology	Typical Speed (operations per second)
1	1946–1957	Vacuum tube	40,000
2	1958–1964	Transistor	200,000
3	1965–1971	Small and medium scale integration	1,000,000
4	1972–1977	Large scale integration	10,000,000
5	1978–1991	Very large scale integration	100,000,000
6	1991-	Ultra large scale integration	1,000,000,000

Moore's Law

- Gordon Moore cofounder of Intel in 1965
- Observed the number of transistors put on a chips doubled every year & predicted this would continue
- True until 70's & slowed to every 18 months and continued ever since

Moore's Law

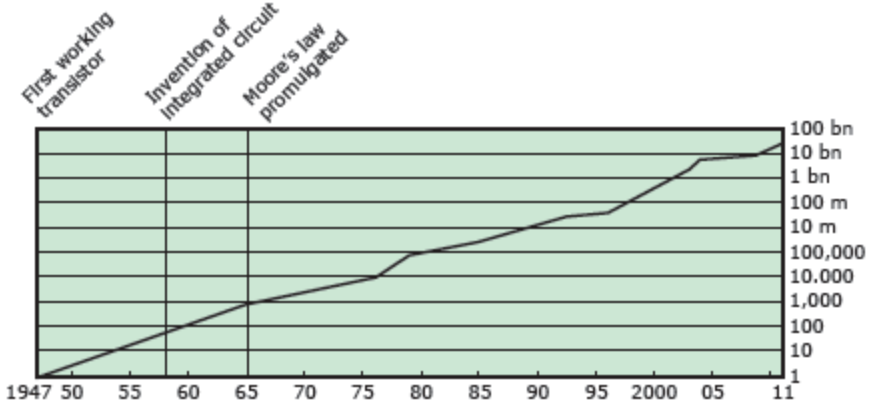


Figure 2.8 Growth in Transistor Count on Integrated Circuits (DRAM memory)

Consequences of Moore's Law

1. Chip cost has remained virtually unchanged meaning the cost computer logic/memory circuitry has dramatically dropped
2. Electrical path shortened increasing operating speed
3. Reduction in power & cooling
4. Integrated circuits mean fewer inter-chip connections using solder therefore more reliable

Evolution of Intel Microprocessors

Table 2.6 Evolution of Intel Microprocessors (page 1 of 2)

(a) 1970s Processors

	4004	8008	8080	8086	8088
Introduced	1971	1972	1974	1978	1979
Clock speeds	108 kHz	108 kHz	2 MHz	5 MHz, 8 MHz, 10 MHz	5 MHz, 8 MHz
Bus width	4 bits	8 bits	8 bits	16 bits	8 bits
Number of transistors	2,300	3,500	6,000	29,000	29,000
Feature size (μm)	10		6	3	6
Addressable memory	640 Bytes	16 KB	64 KB	1 MB	1 MB

Evolution of Intel Microprocessors

(b) 1980s Processors

	80286	386TM DX	386TM SX	486TM DX CPU
Introduced	1982	1985	1988	1989
Clock speeds	6 MHz - 12.5 MHz	16 MHz - 33 MHz	16 MHz - 33 MHz	25 MHz - 50 MHz
Bus width	16 bits	32 bits	16 bits	32 bits
Number of transistors	134,000	275,000	275,000	1.2 million
Feature size (µm)	1.5	1	1	0.8 - 1
Addressable memory	16 MB	4 GB	16 MB	4 GB
Virtual memory	1 GB	64 TB	64 TB	64 TB
Cache	—	—	—	8 kB

Evolution of Intel Microprocessors

(c) 1990s Processors

	486TM SX	Pentium	Pentium Pro	Pentium II
Introduced	1991	1993	1995	1997
Clock speeds	16 MHz - 33 MHz	60 MHz - 166 MHz,	150 MHz - 200 MHz	200 MHz - 300 MHz
Bus width	32 bits	32 bits	64 bits	64 bits
Number of transistors	1.185 million	3.1 million	5.5 million	7.5 million
Feature size (µm)	1	0.8	0.6	0.35
Addressable memory	4 GB	4 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	8 kB	8 kB	512 kB L1 and 1 MB L2	512 kB L2

Evolution of Intel Microprocessors

(d) Recent Processors

	Pentium III	Pentium 4	Core 2 Duo	Core i7 EE 990
Introduced	1999	2000	2006	2011
Clock speeds	450 - 660 MHz	1.3 - 1.8 GHz	1.06 - 1.2 GHz	3.5 GHz
Bus width	64 bits	64 bits	64 bits	64 bits
Number of transistors	9.5 million	42 million	167 million	1170 million
Feature size (nm)	250	180	65	32
Addressable memory	64 GB	64 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	512 kB L2	256 kB L2	2 MB L2	1.5 MB L2/12 MB L3

Rest of Chapter

- Read 2.2, 2.3, 2.4, and 2.5 on your own. You don’t have to understand every little detail as the rest of the course will cover info in these sections in great detail.
- I will cover 2.6 at a later date