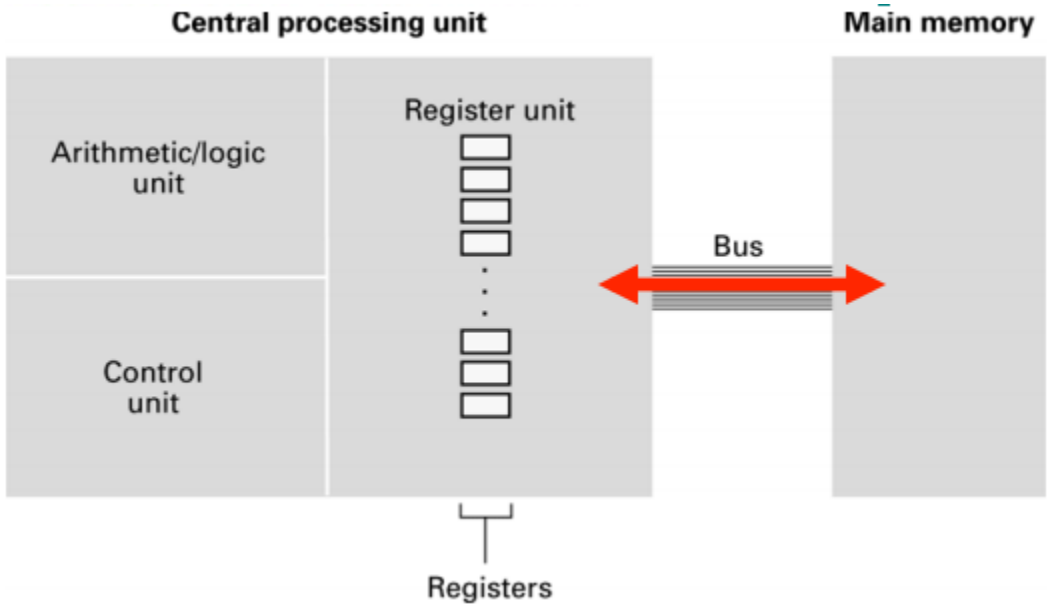
How Computers Calculate

What is inside?



What does the user see? what the computer does?

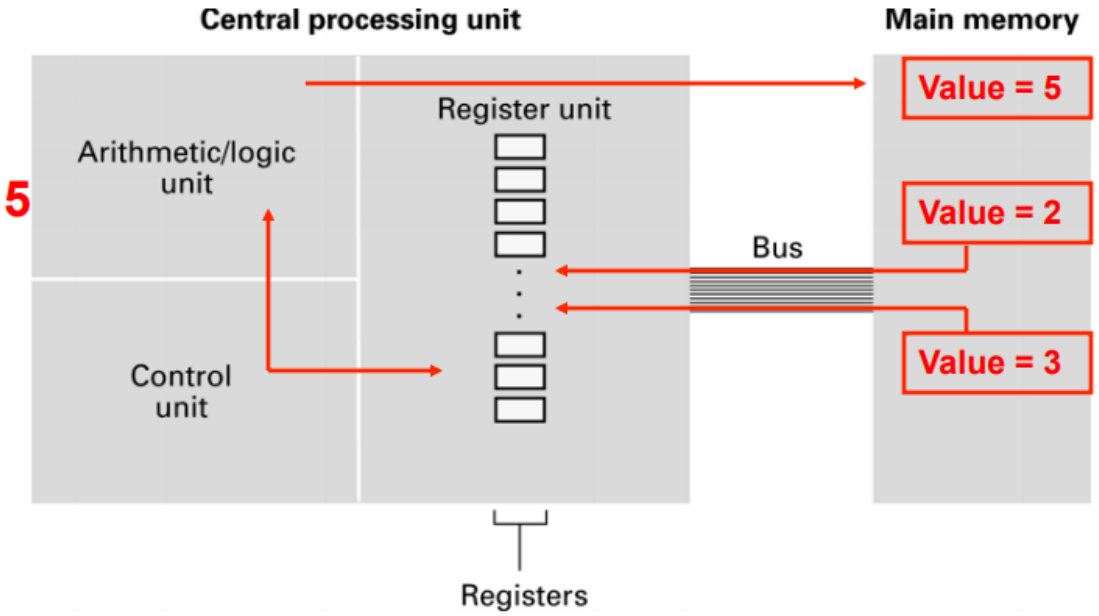
User Types (Input)

Computer Outputs

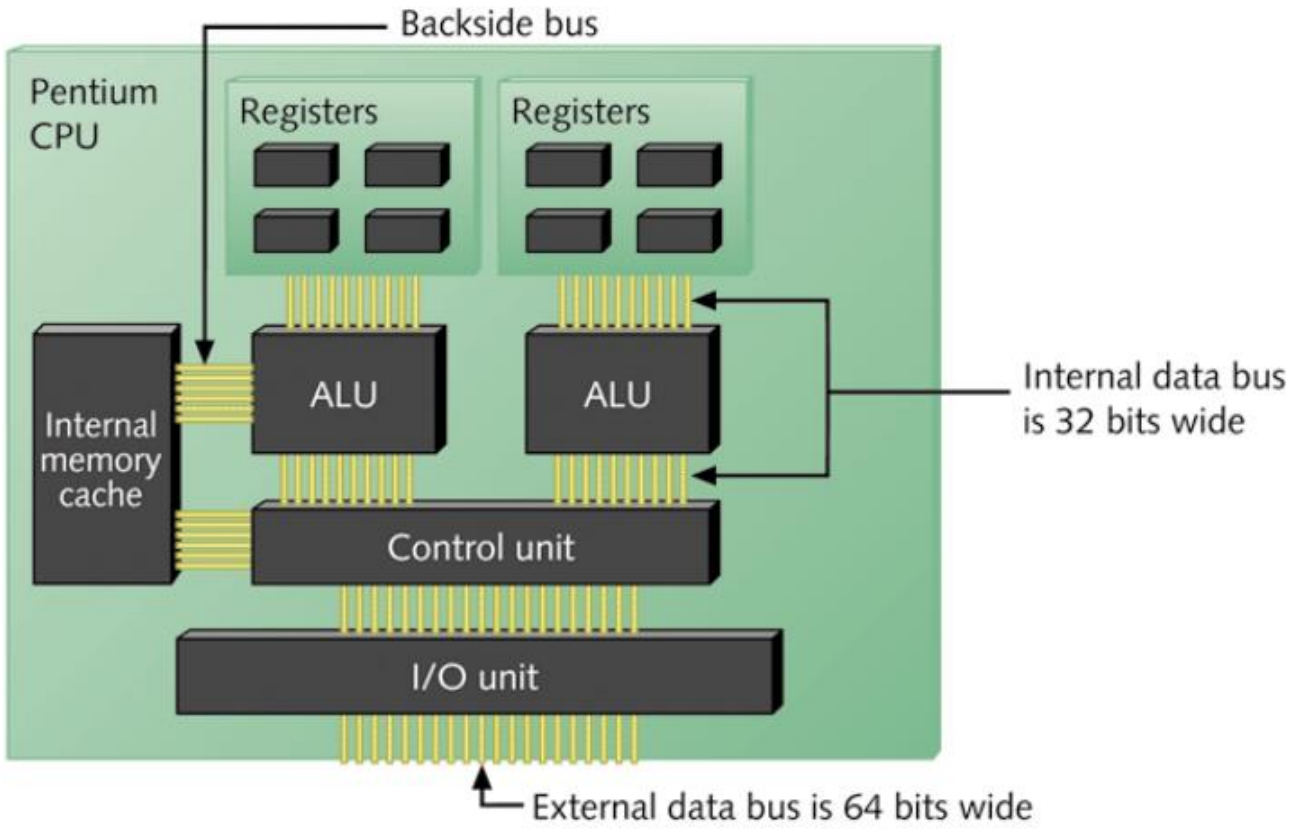
2 + 3 =

5

2 + 3 = 5



CPU


Backside bus
Pentium CPU
Registers
Registers
ALU
ALU
Internal memory cache
Internal data bus is 32 bits wide
Control unit
I/O unit
External data bus is 64 bits wide

Adding values stored in memory

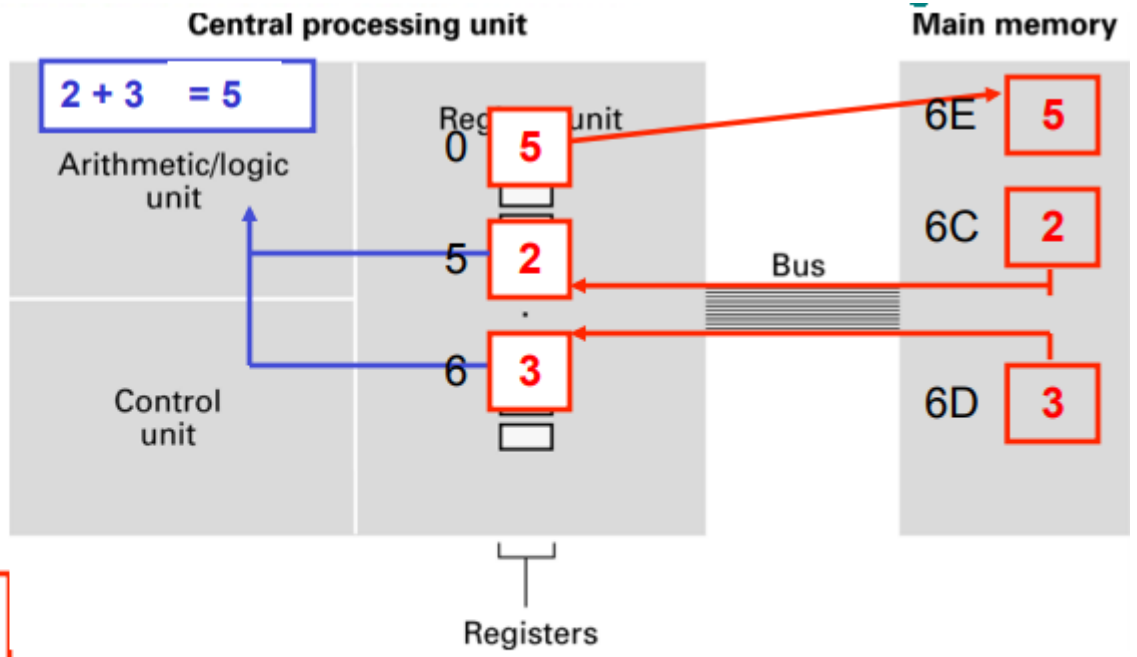
Step 1 Load register 5 with the bit pattern found in the memory cell at address 6C.

Step 2 Load register 6 with the bit pattern found in the memory cell at address 6D.

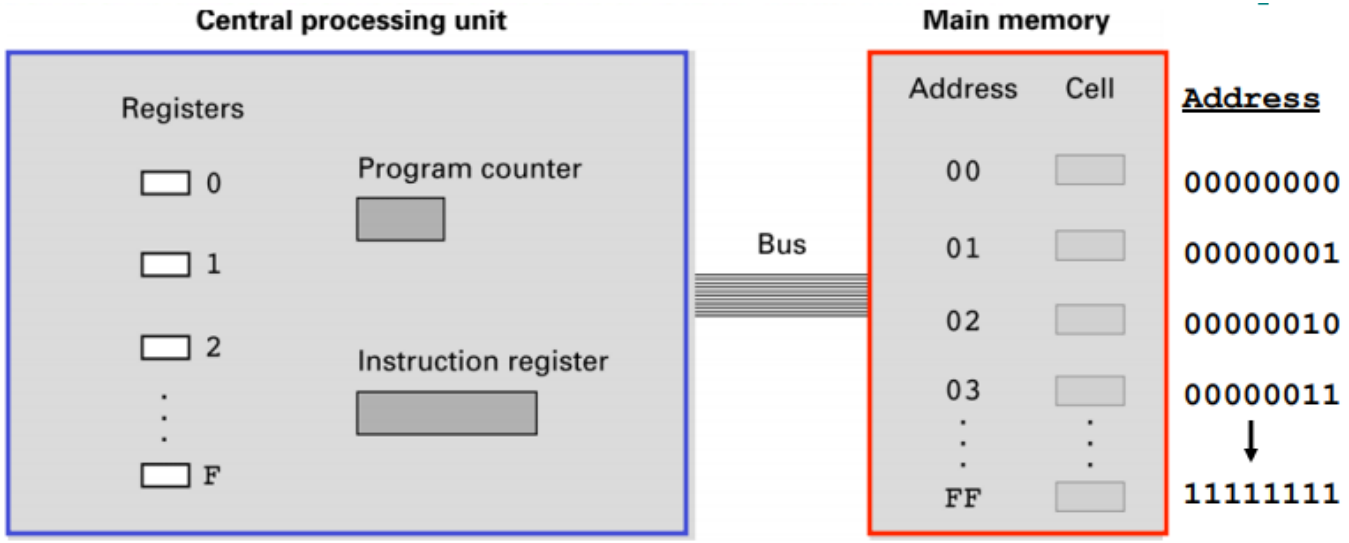
Step 3 Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.

Step 4 Store the contents of register 0 in the memory cell at address 6E.

Step 5 Halt.



The architecture of our machine



- **Main Memory**
  - 256 cells: 00 through FF (Hex)
    - 0000 0000 through 1111 1111
  - Storage: 8 bits per cell
- **CPU**
  - **16 registers**: 0 through F (Hex); 8 bits per cell
  - **Program counter**: (Address of) Keeps track of the next instruction
  - **Instruction register**: Contains the current instruction to be executed by the ALU.

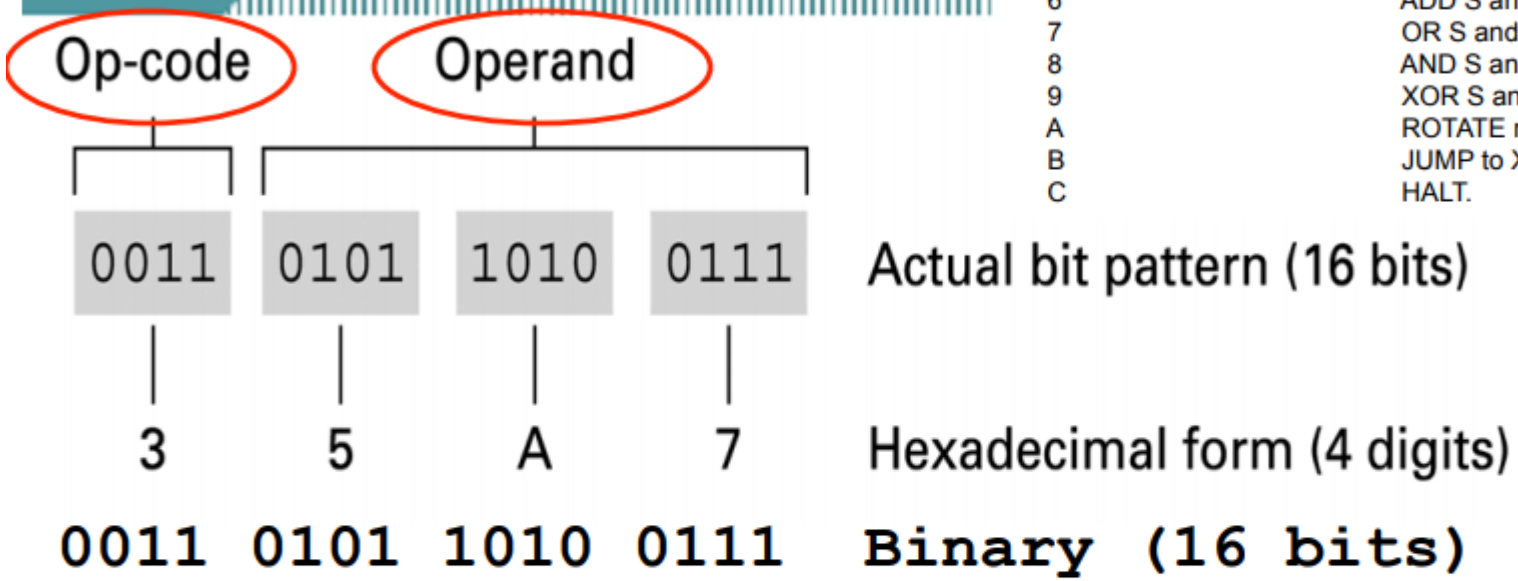
Converting Decimal, Hex, and Binary

Dec.	Hex.	Binary	Dec.	Hex.	Binary
0	0	0000	8	8	1000
1	1	0001	9	9	1001
2	2	0010	10	A	1010
3	3	0011	11	B	1011
4	4	0100	12	C	1100
5	5	0101	13	D	1101
6	6	0110	14	E	1110
7	7	0111	15	F	1111

1 Hex digit = 4 bits

Parts of a Machine Instruction

Op-code	Operand	Description
1		LOAD reg. R from cell XY.
2		LOAD reg. R with XY.
3		STORE reg. R at XY.
4		MOVE R to S.
5		ADD S and T into R. (2's comp.)
6		ADD S and T into R. (floating pt.)
7		OR S and T into R.
8		AND S and T into R.
9		XOR S and T into R.
A		ROTATE reg. R X times.
B		JUMP to XY if R = reg. 0.
C		HALT.



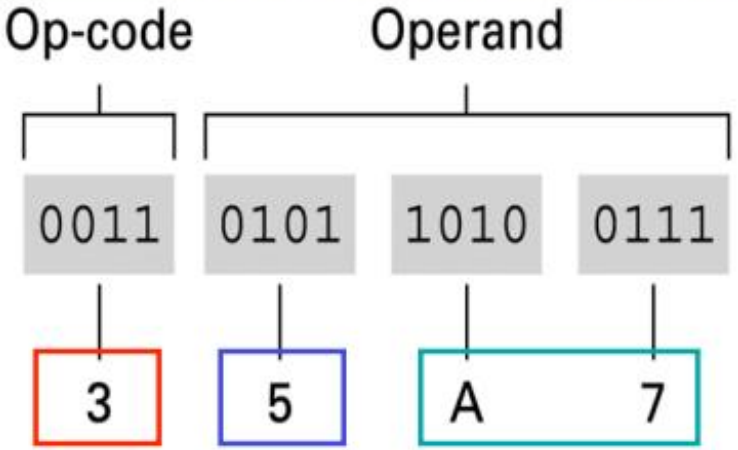
1 Hex digit = 4 bits

Machine Language

- Each instruction involves two parts:
  - **Op-code:** Specifies which operation to execute
    - LOAD, ADD, STORE, etc.
  - **Operand:** Gives more detailed information about the operation
    - Interpretation of operand varies depending on op-code

Parts of a Machine Instruction

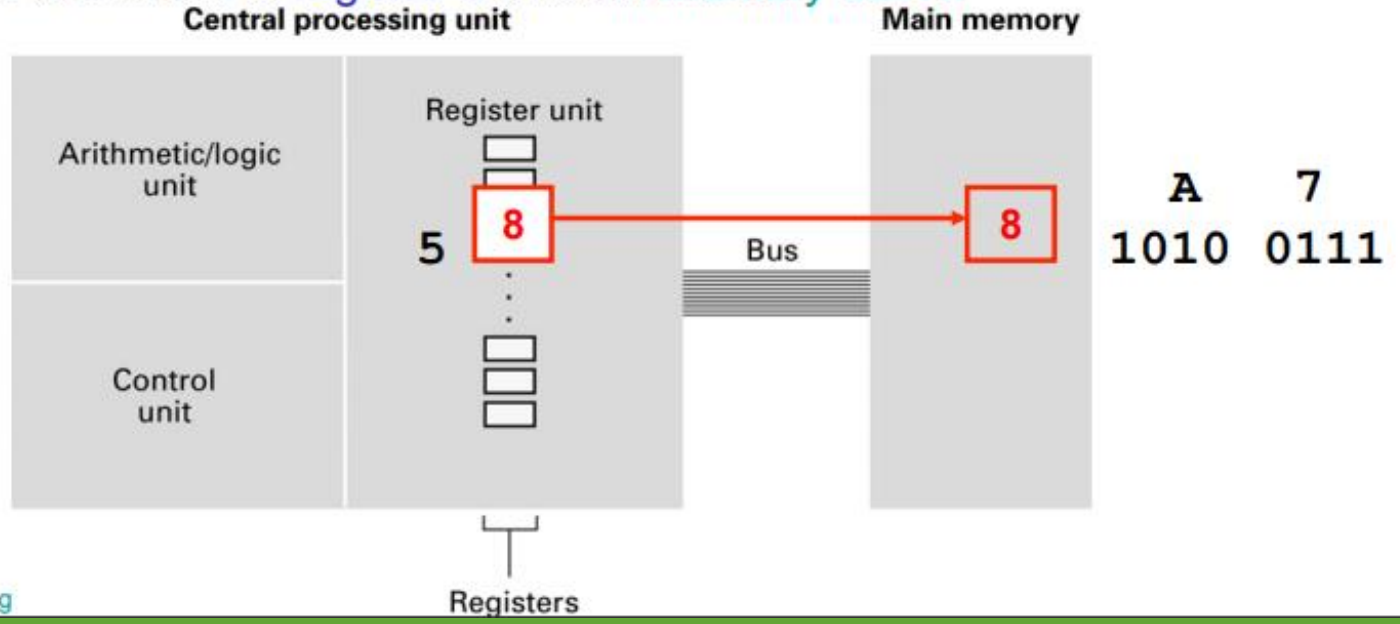
Op-code	Operand	Description
1		LOAD reg. R from cell XY.
2		LOAD reg. R with XY.
3		STORE reg. R at XY.
4		MOVE R to S.
5		ADD S and T into R. (2's comp.)
6		ADD S and T into R. (floating pt.)
7		OR S and T into R.
8		AND S and T into R.
9		XOR S and T into R.
A		ROTATE reg. R X times.
B		JUMP to XY if R = reg. 0.
C		HALT.



Actual bit pattern (16 bits)

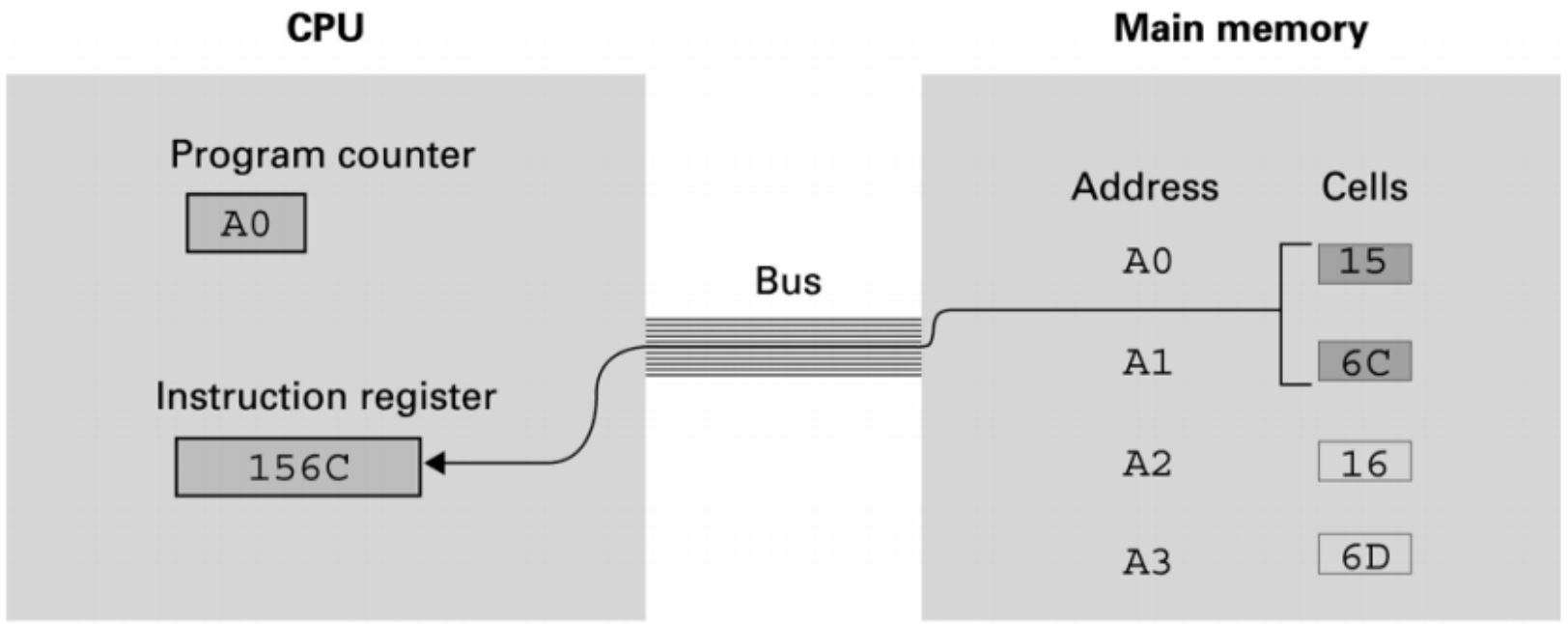
Hexadecimal form (4 digits)

Store the bits found in register 5 in main memory cell A7



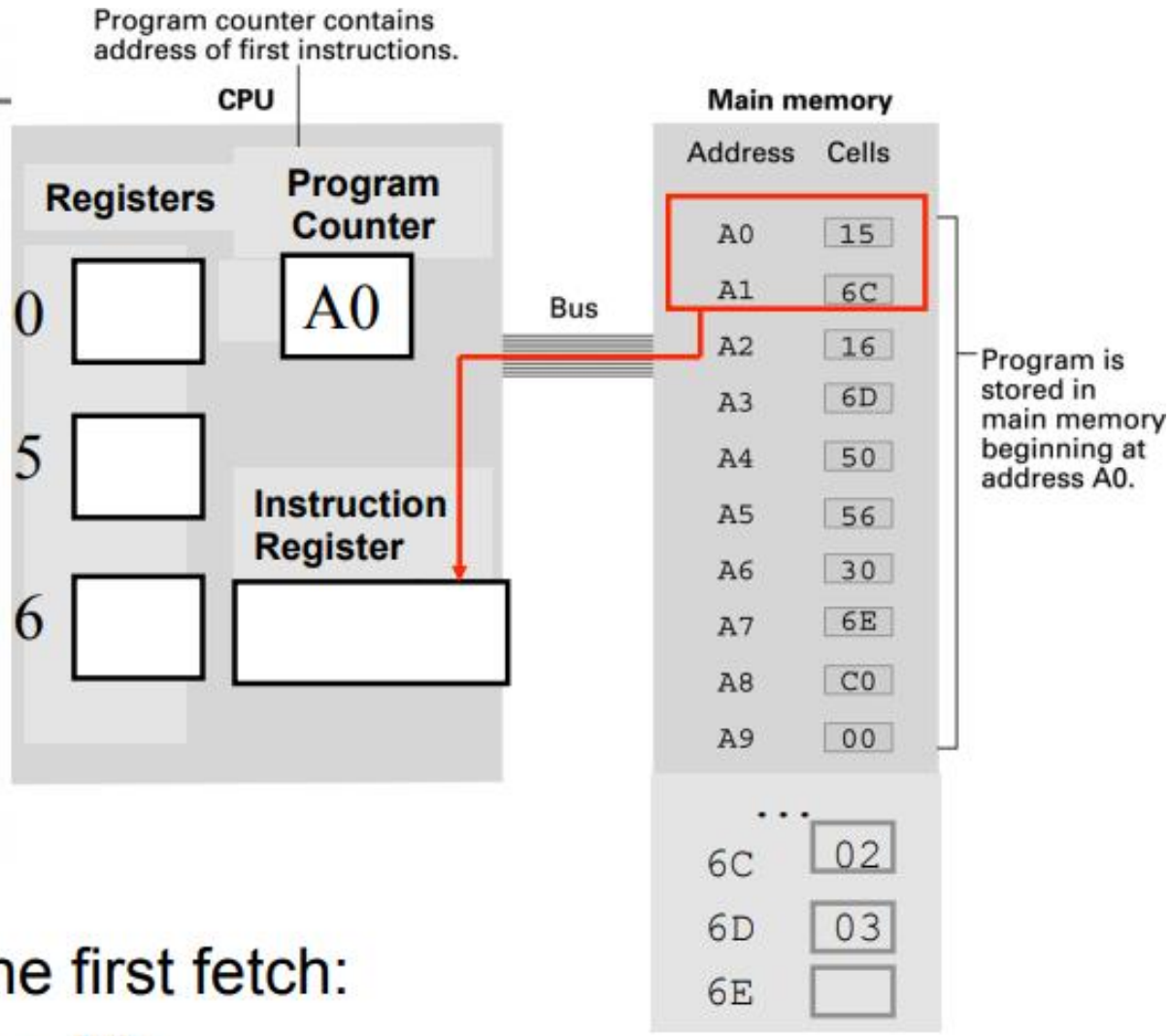
A 7
1010 0111

Performing the fetch step of the machine cycle



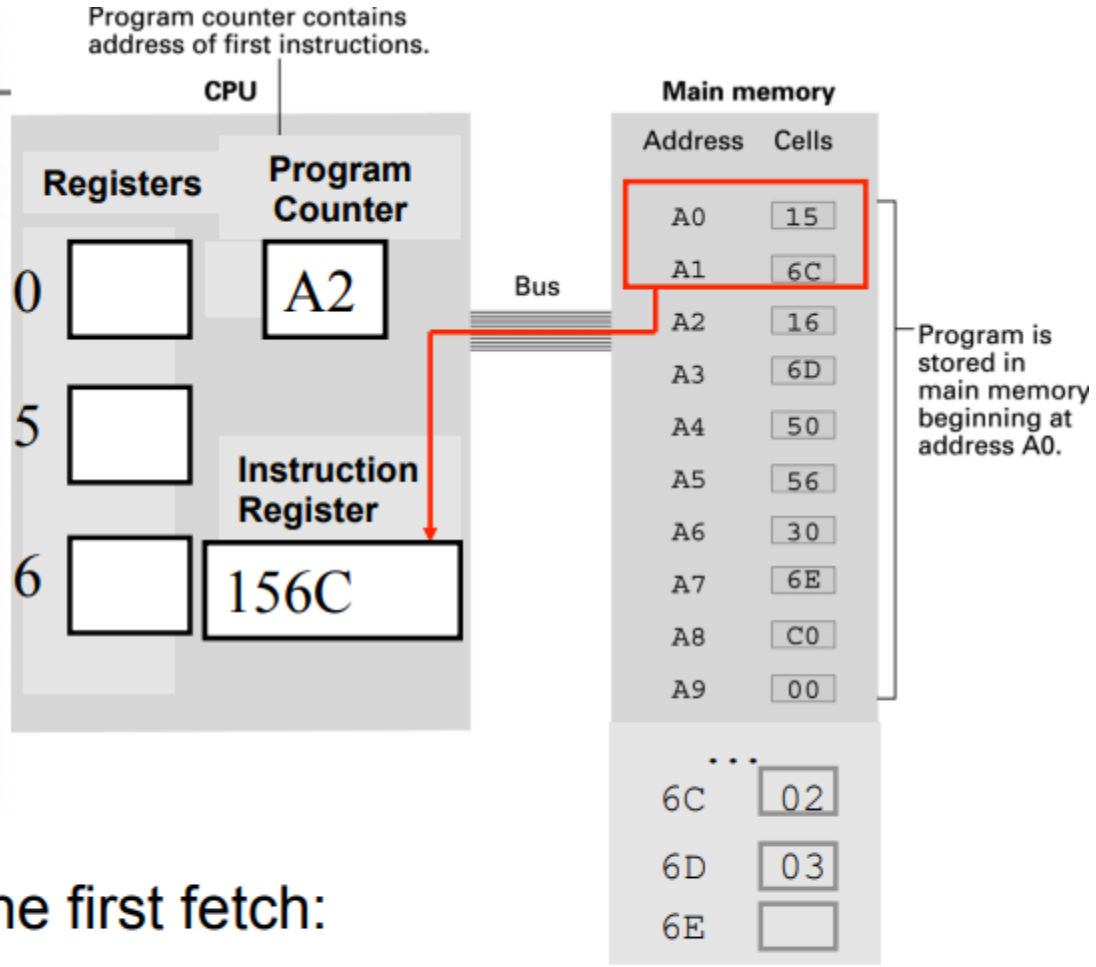
a. At the beginning of the fetch step the instruction starting at address A0 is retrieved from memory and placed in the instruction register.

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.

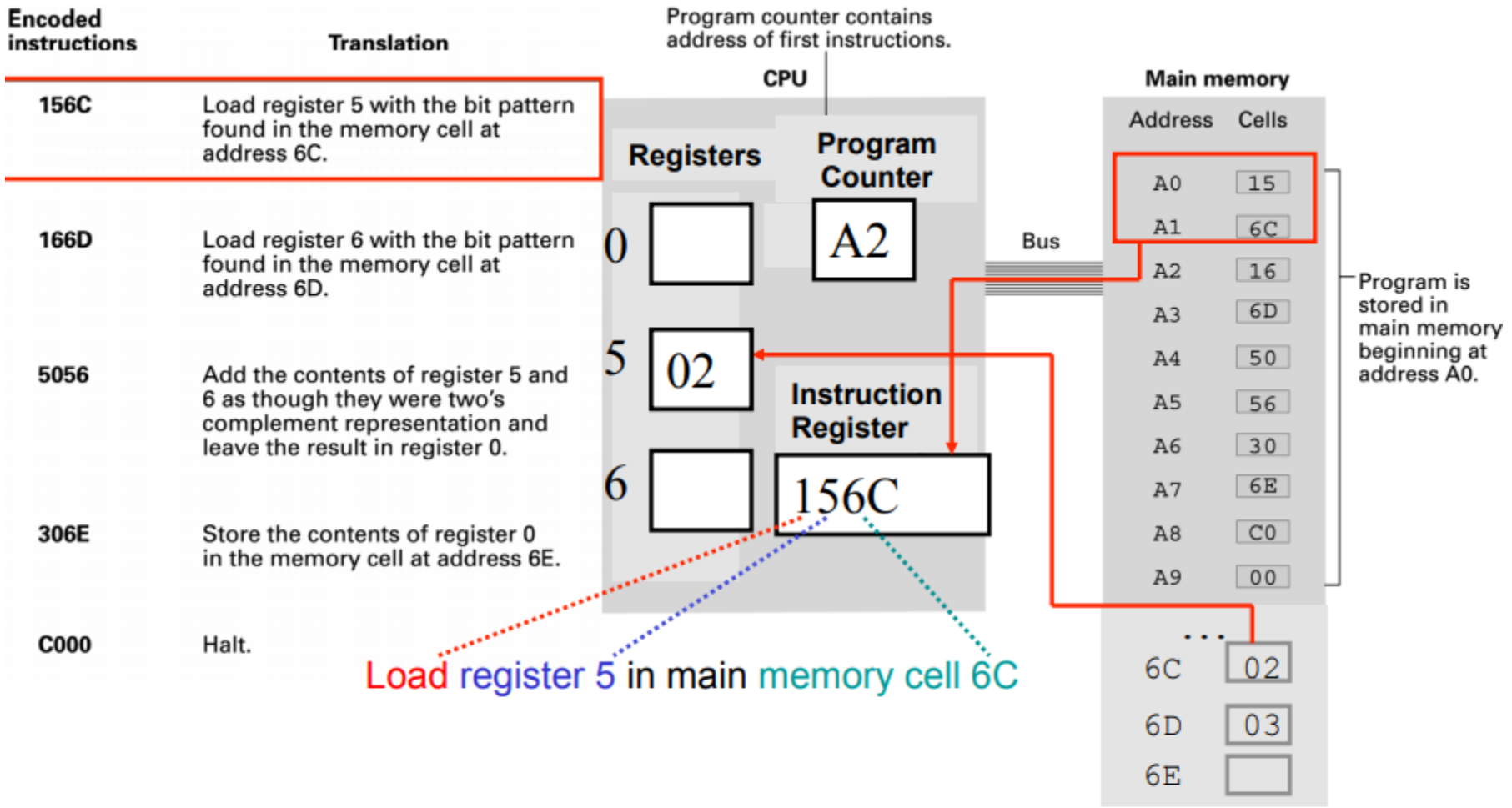


- At the beginning of the first fetch:
  - Program Counter = A0

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.

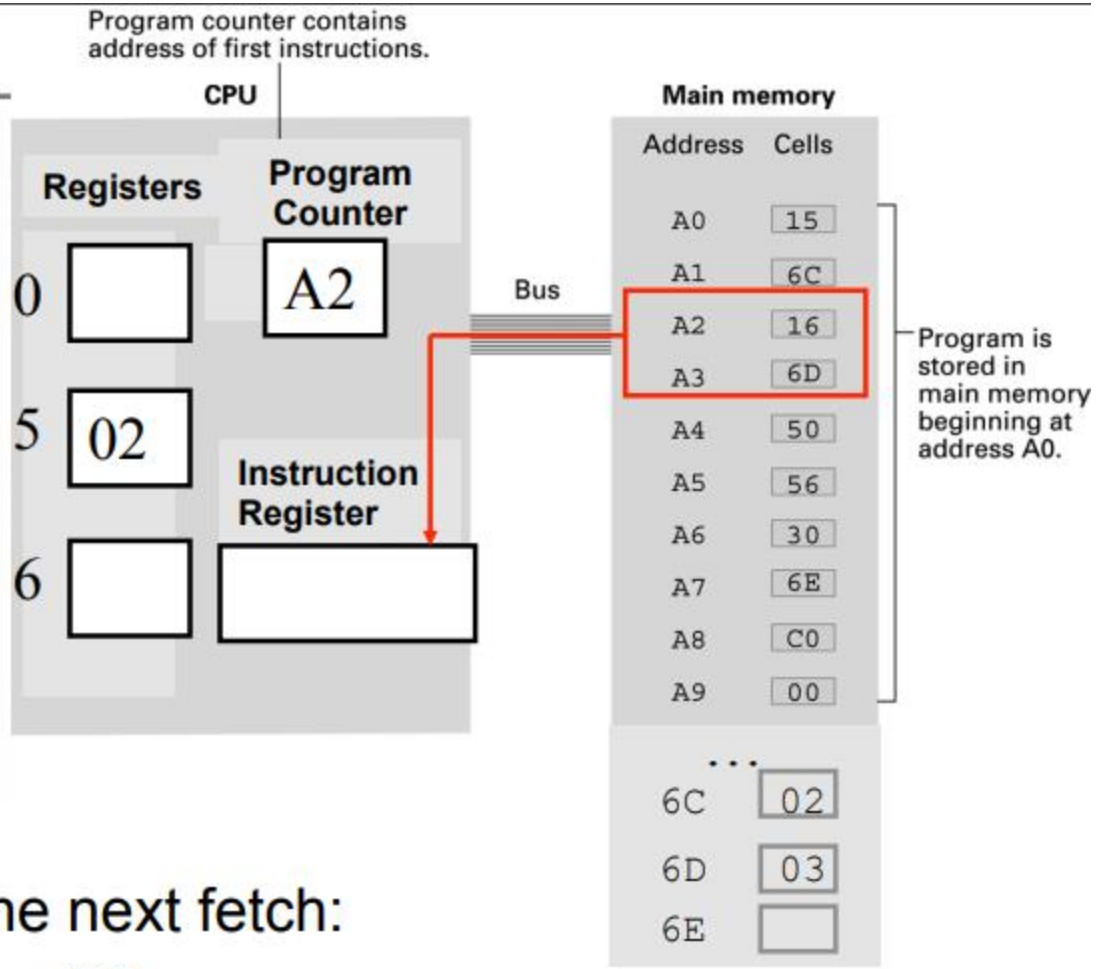


- At the beginning of the first fetch:
  - Instruction at A0 (and A1) loaded into Instruction register
  - Program Counter = A2



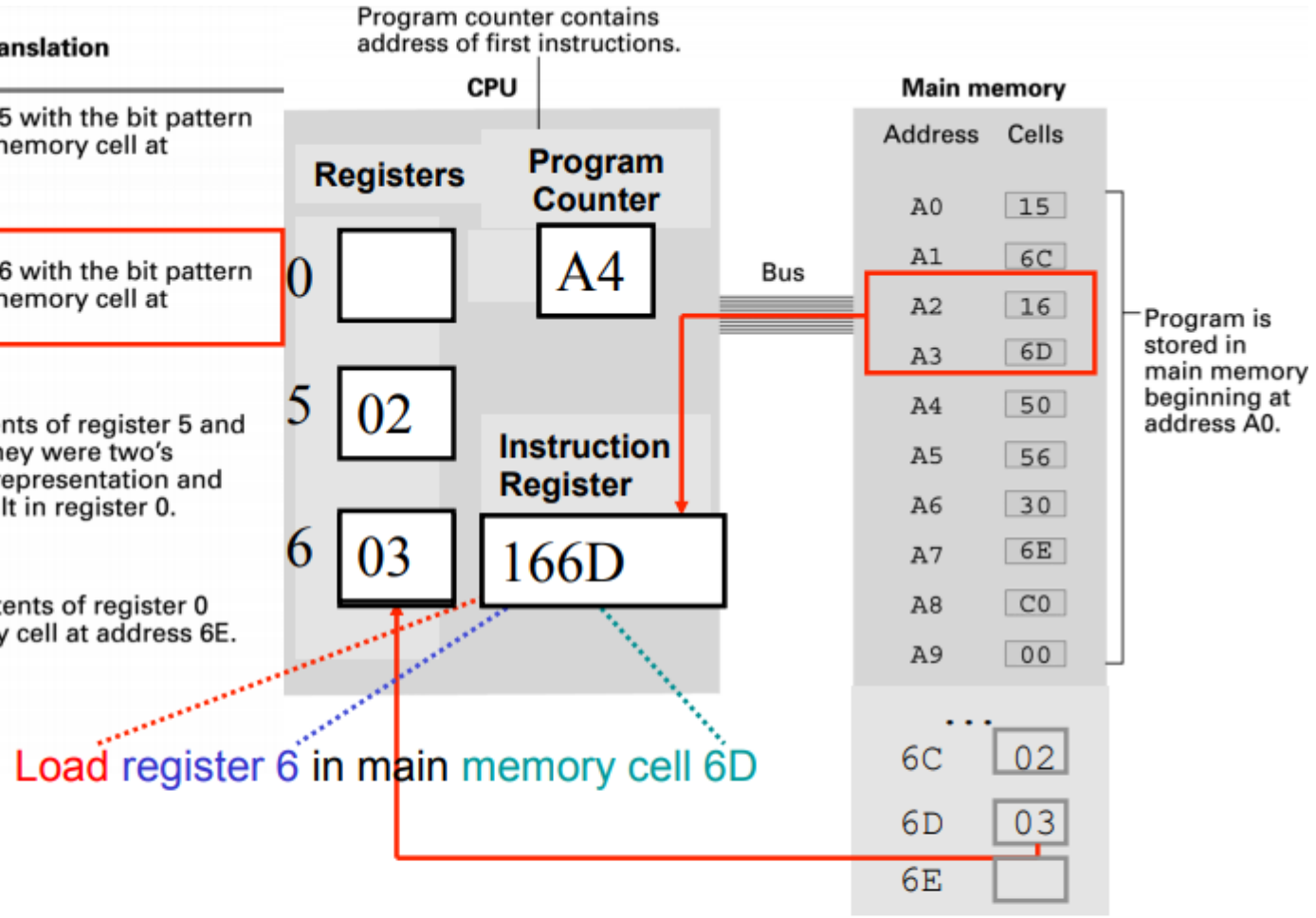
- CPU analyzes the instruction
- Loads Register 5 with the contents of memory cell address 6C.

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



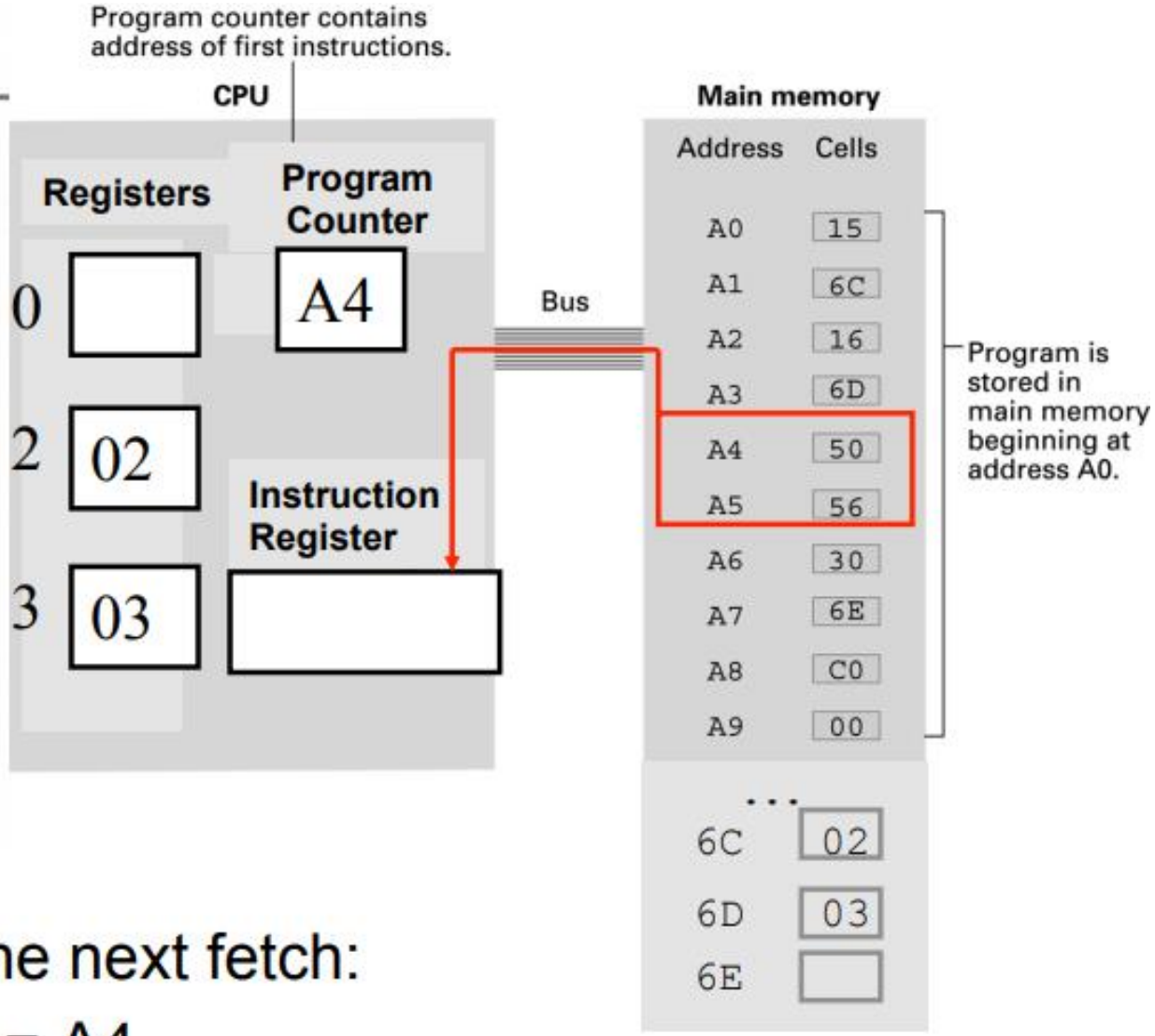
- At the beginning of the next fetch:
  - Program Counter = A2

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



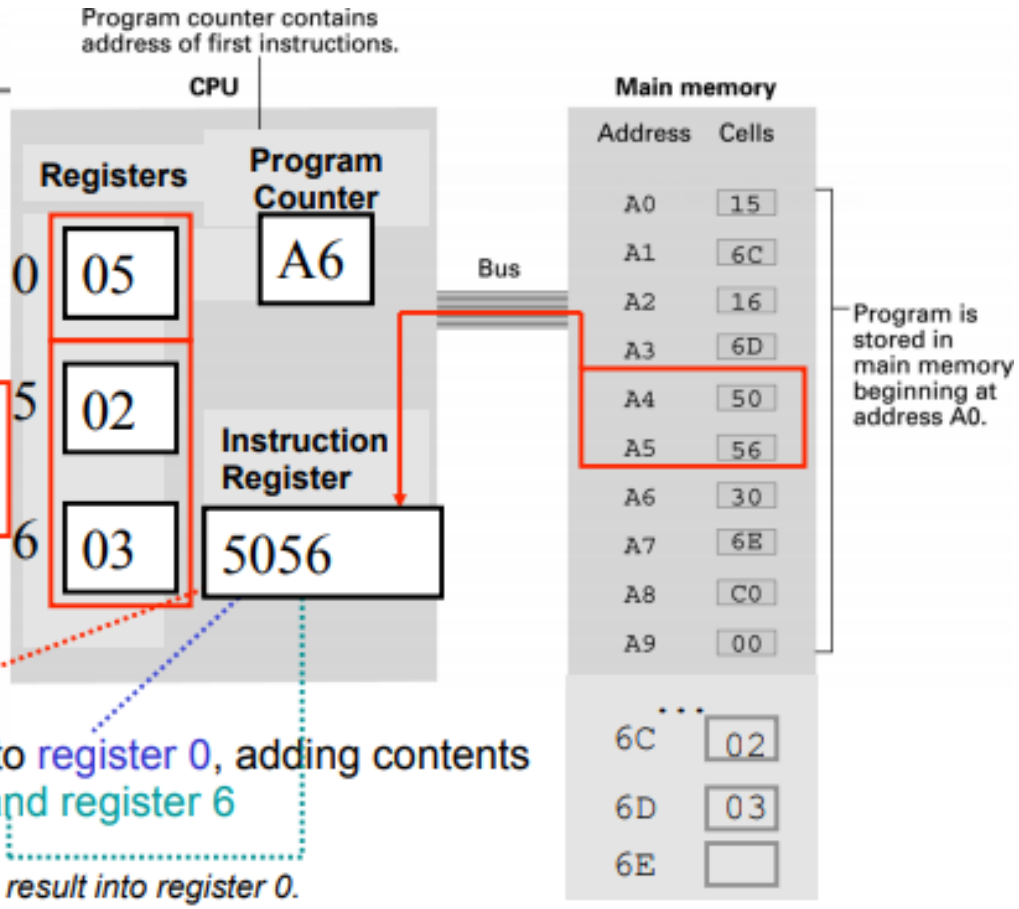
- Instruction at A2 (and A3) loaded into Instruction register
- Program Counter incremented: A4
- CPU analyzes the instruction
- Loads Register 6 with the contents of memory cell address 6D.

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



- At the beginning of the next fetch:
  - Program Counter = A4

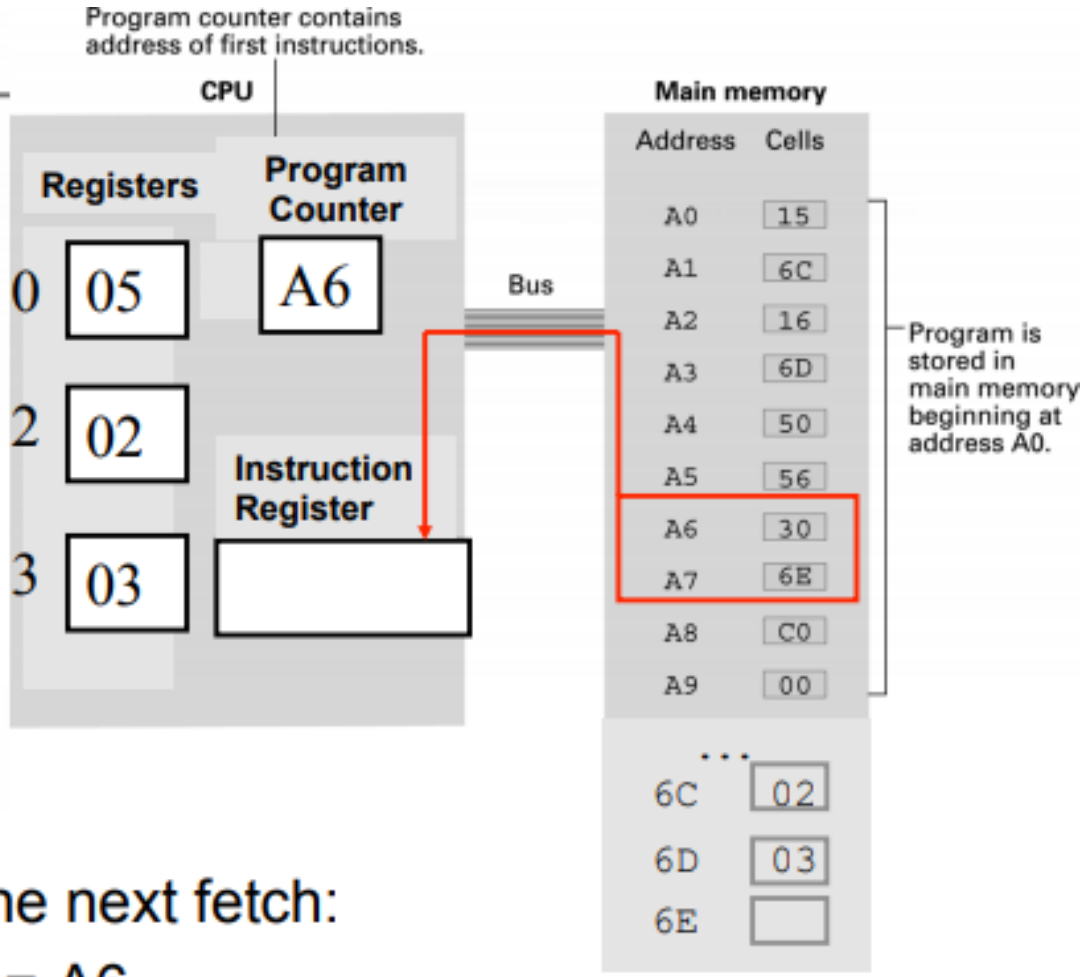
Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



Adds contents of register 5 and 6, placing result into register 0.

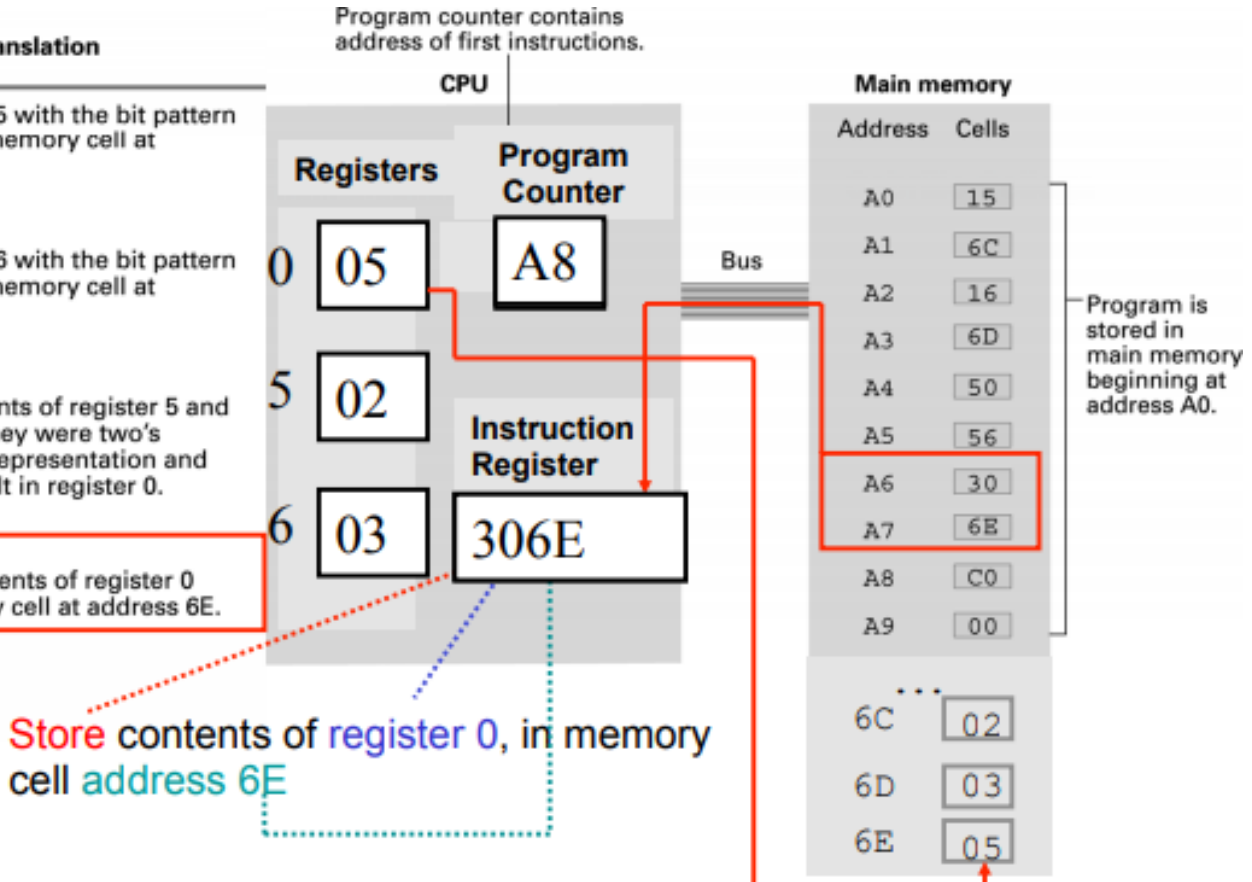
- Instruction at A4 (and A5) loaded into Instruction register
- Program Counter incremented: A6
- CPU analyzes the instruction
- Adds contents of register 5 and register 6, storing the result into register 0.

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



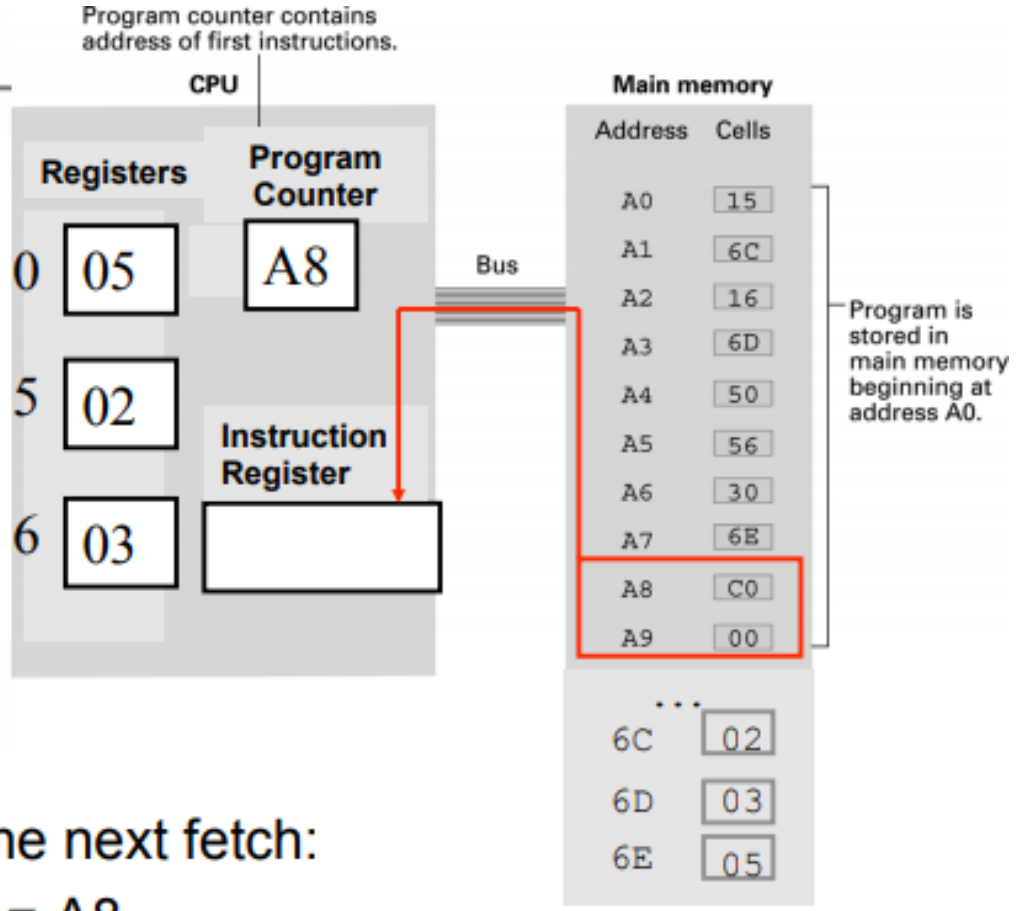
- At the beginning of the next fetch:
  - Program Counter = A6

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



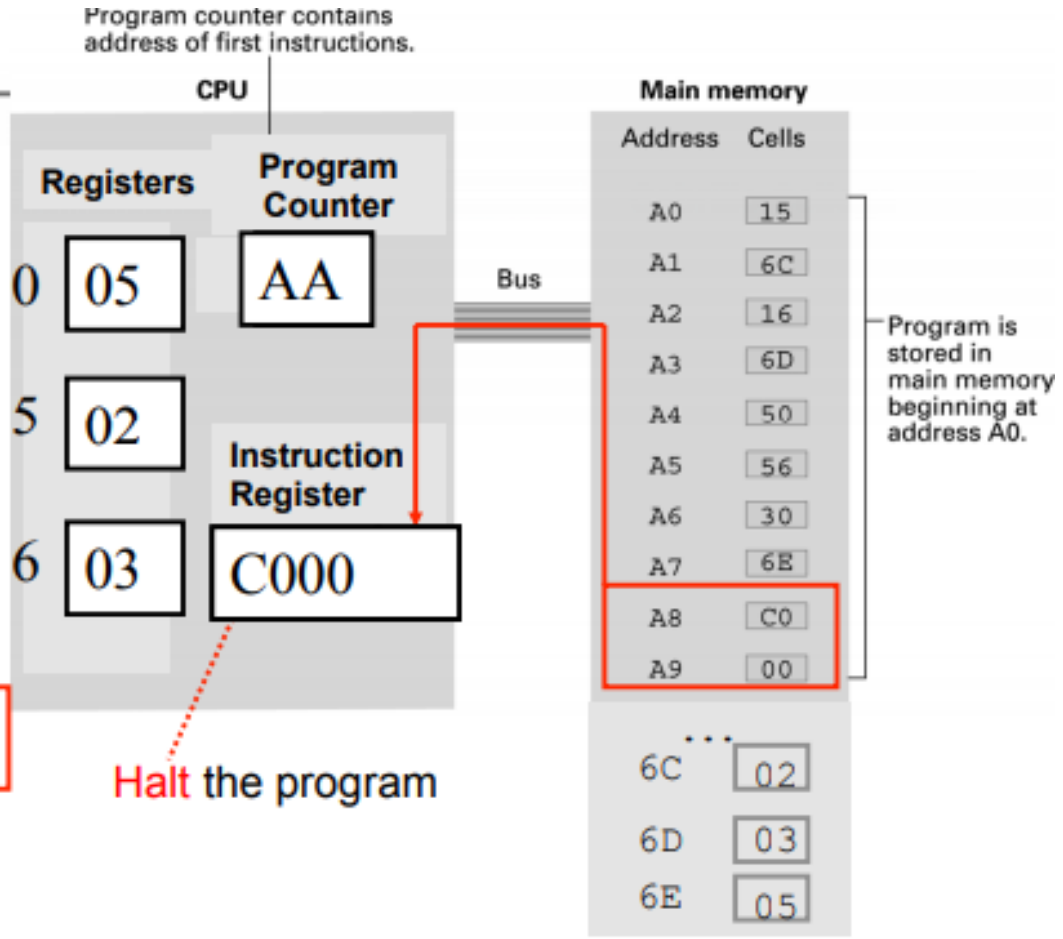
- Instruction at A6 (and A7) loaded into Instruction register
- Program Counter incremented: A8
- CPU analyzes the instruction
- Stores contents of register 0 in the memory cell at address 6E.

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



- At the beginning of the next fetch:
  - Program Counter = A8

Encoded instructions	Translation
156C	Load register 5 with the bit pattern found in the memory cell at address 6C.
166D	Load register 6 with the bit pattern found in the memory cell at address 6D.
5056	Add the contents of register 5 and 6 as though they were two's complement representation and leave the result in register 0.
306E	Store the contents of register 0 in the memory cell at address 6E.
C000	Halt.



- Instruction at A8 (and A9) loaded into Instruction register
- Program Counter incremented: AA
- CPU analyzes the instruction
- Halts the program. (Program ends.)