Microprocessors, Lecture 3

Z80 Instruction Set

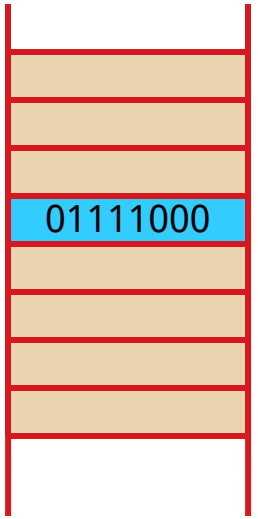
Hamid Fadishei
Assistant Professor, University of Bojnord
Spring 2015

Z80 instruction set

- Z80 instruction set
  - 158 instruction types
  - 6 major categories
    - Data transfer (load) operations
    - Arithmetic operations
    - Logic operations
    - Bit manipulation
    - Branch operations
    - Machine control operations
- Instruction parts
  - Operation code (opcode)
  - Data to be operated on
- The size of opcode part is usually 1 (and sometimes 2)
- The size of the instruction varies between 1 and 4 bytes

Z80 instruction set

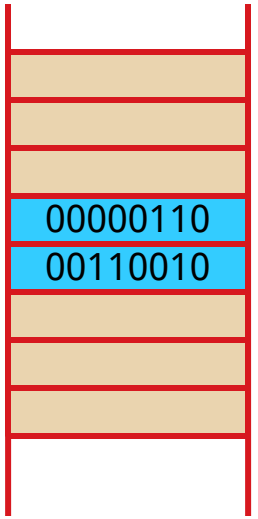
- 1-byte instructions
  - Opcode and operand in the same byte



Opcode	Operand	Binary code	Task
LD	A, B	01111000 (78H)	A ← B
ADD	A, B	10000000 (80H)	A ← A + B

Z80 instruction set

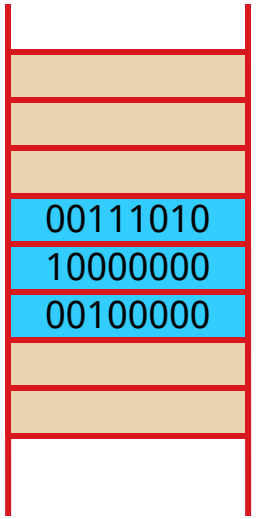
- 2-byte instructions
  - First byte: opcode
  - Second byte: operand



Opcode	Operand	Binary code	Task
LD	B, 32H	00000110 (06H) 00110010 (32H)	B ← 32H

Z80 instruction set

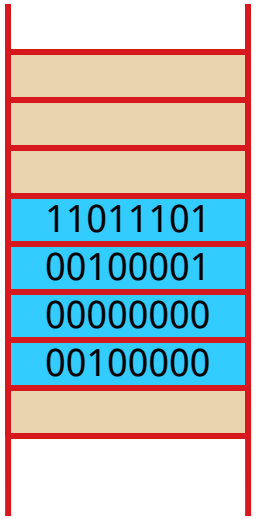
- 3-byte instructions
  - First byte: opcode
  - Second and third bytes: 16-bit address or data



Opcode	Operand	Binary code	Task
LD	A, (2080H)	00111010 (3AH) 10000000 (80H) 00100000 (20H)	A ← MEM[2080]

Z80 instruction set

- 4-byte instructions
  - First and second bytes: opcode
  - Third and forth bytes: operand



Opcode	Operand	Binary code	Task
LD	IX, 2000H	11011101 (DDH) 00100001 (21H) 00000000 (00H) 00100000 (20H)	IX ← 2000H

Data transfer (load) operations

- Copy date from “source” to “destination”
- Do not affect flags

Transfer type	Example	Task
From register to register	LD A, B	A ← B
8-bit data to register	LD B, 25H	B ← 35H
16-bit data to register (pair)	LD HL, 2050H	HL ← 2050H
From memory location to register	LD A, (2080H)	A ← MEM[2080H]
From register to memory location	LD (2040H), A	MEM[2080H] ← A
From input port to accumulator	IN A, (01H)	A ← INP[01H]
From accumulator to output port	OUT (02H), A	OUT[02H] ← A
From register pair to stack memory	PUSH BC	MEM[SP-1] ← B MEM[SP-2] ← C SP ← SP-2
From stack to register pair	POP DE	E ← MEM[SP] D ← MEM[SP+1] SP ← SP+2
Exchange (swap)	EXX	BC ↔ BC’ DE ↔ DE’ HL ↔ HL’

Arithmetic operations

- Addition
  - Can be added to the content of accumulator:
    - Content of a register
    - Any 8-bit number
    - Content of a memory location
  - You can not add to other registers directly
- Subtraction
  - Can be subtracted from the content of accumulator:
    - Content of a register
    - Any 8-bit number
    - Content of a memory location
- Affect flags

Arithmetic operation	Example	Task
Add register to accumulator	ADD A, B	A ← A + B
Add 8-bit data to accumulator	ADD A, 25H	A ← A + 25H
Add memory content to accumulator	ADD A, (HL)	A ← A + MEM[HL]
Subtract register from accumulator	SUB C	A ← A - C

Arithmetic operations

- Increment/decrement
  - Can be incremented or decremented by 1:
    - Content of an 8-bit register
    - Content of an 8-bit memory location
    - Content of an 16-bit register pair
  - Affects flags except carry for 8-bit operation
  - Does not affects flags for 16-bit operation
- 1's and 2's complement
  - Operate on accumulator
  - Affect some flags

Arithmetic operation	Example	Task
Increment register content	INC B	B ← B + 1
Decrement register pair content	DEC BC	BC ← BC - 1
1's complement	CPL	A ← NOT(A)
2's complement	NEG	A ← -A

Arithmetic operations

- Comparison
  - Actually a subtraction
  - But the result will not be stored in accumulator
  - Only affects flags

Comparison operation	Example	Task
Compare accumulator with register	CP B	A - B
Compare accumulator with a number	CP 25H	A – 25H

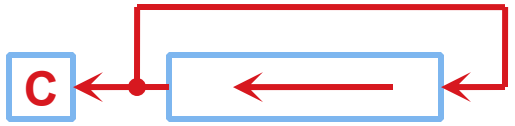
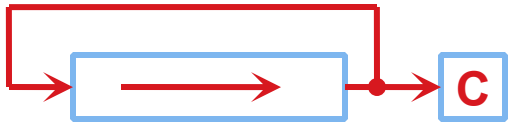
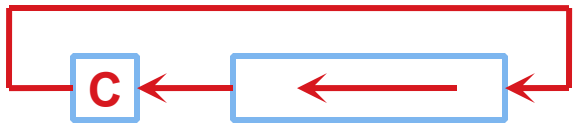
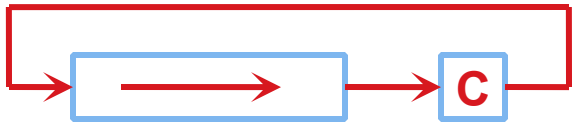
Logic operations

- Logic operations
  - Logic functions (AND, OR, XOR)
  - Shift and rotate

Logic operation	Example	Task
Logical AND	AND B	A ← A AND B
Logical XOR	XOR C	A ← A XOR C
Logical OR	OR (HL)	A ← A OR MEM[HL]
Rotate Left the content of a register (Along with carry)	RL A	Refer to the next slide
Rotate Right the content of a register (Along with carry)	RR B	Refer to the next slide

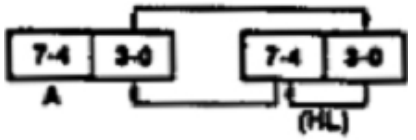
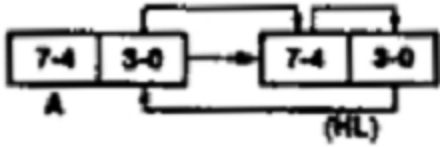
Logic operations

- Various shift operations

Shift operation	Mnemonic	Diagram
Rotate left	RLC	
Rotate right	RRC	
Rotate left along with Carry	RL	
Rotate right along with Carry	RR	

Logic operations

- Various shift operations

Shift operation	Mnemonic	Diagram
Shift left arithmetic	SLA	
Shift right logical	SRL	
Shift right arithmetic	SRA	
Rotate left decimal	RLD	
Rotate right decimal	RRD	

Bit Manipulation

- Set/Reset individual bits of registers or memory locations
  - Test bits of registers or memory locations
  - Result goes to Z flag

Bit manipulation operation	Example	Task
Set a bit in a register	SET 5, A	Bit D_5 of A becomes 1
Reset a bit in a register	RES 4, B	Bit D_4 of B becomes 0
Test a bit in a register	BIT 6, B	Test whether bit D_6 of B is 1 or 0

Branching operations

- Jump
  - Conditional/Unconditional
  - Absolute/Relative
- Call/Return
  - Conditional/Unconditional
- Restart
  - Call special locations in page 00 (more on this later)

Branch operation	Example	Task
Jump to an absolute address unconditionally	JP 2030H	PC ← 2030H
Jump to an absolute address conditionally	JP C, 2030H	If (C=1) then PC ← 2030H
Jump to a relative address unconditionally	JR 15	PC ← PC + 15
Jump to a relative address conditionally	JR Z, 25	If (Z=1) then PC ← PC + 25
Call a subroutine	CALL 1020H	MEM[SP-1] ← PC_H MEM[SP-2] ← PC_L PC ← 1020H
Return from a subroutine	RET	PC_L ← MEM[SP] PC_H ← MEM[SP+1]

Machine control operations

- Control various CPU operations
  - Stop execution of instructions (HALT)
  - Enable/Disable interrupts (EI/DI)

Z80 addressing modes

- Addressing mode
  - The way of specifying an operand or a data location
  - Z80 has 10 addressing modes
  - You have already seen some
    - Immediate
      - ADD A, 25H
    - Register
      - LD B, C

Z80 addressing modes

Addressing mode	Explanation	Example(s)
Immediate	The byte following the opcode is the operand	LD B, 56H
Immediate extended	The two bytes following the opcode make the 16-bit operand	LD BC, 20F0H
Register	The operand is one of the registers which is mentioned as a part of the opcode	LD A, B
Implied	The operand is implied by the instruction opcode	AND B
Register indirect	The operand is in a memory location addressed by a 16-bit register pair	LD B, (HL) JP (HL)
Extended	The address of the operand is in the instruction itself	LD A, (2070H)
Relative	The second byte specifies a 2's complement displacement value (added to PC)	JR 15
Indexed	The second byte specifies a 2's complement displacement value (added to IX or IY)	LD (IY+12), 20H INC (IX+20)
Bit	A bit from a register or memory location is specified in the instruction	SET 7, B
Page zero	The 16-bit operand consists of the specified 8-bit value in the high order and a 00 in the high order bits	RST 28H