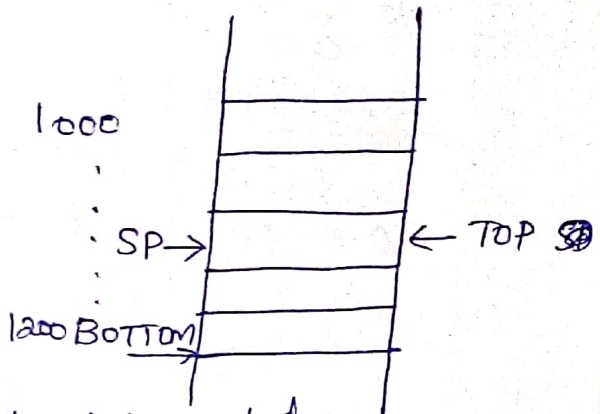
June 10, 2020

COA
Series 1

KOWSIK NANDA GOPAND
ROLL NO 31 CSE S4
MOB: 7356823609

2.

~~Assume~~ Let the diagram shown represent the the bottom be the. initial address of the stack
The stack fills in such a way that it decrease in address value.



STACK PUSH

MOVE R0, -(SP)

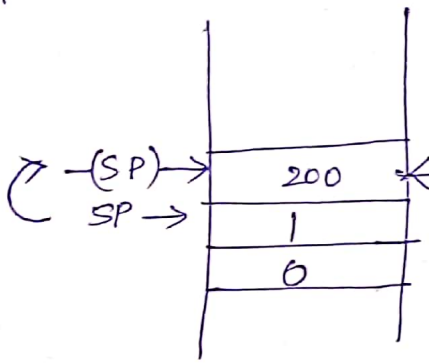
Let R0 contain value to be pushed and SP represent the initial pointed location.

Alterate Method PUSH
Sub #4 SP
MOV R0 (SP)

STACK POP

MOVE (SP)+, R0

For example if value 200 is in R0 when pushed

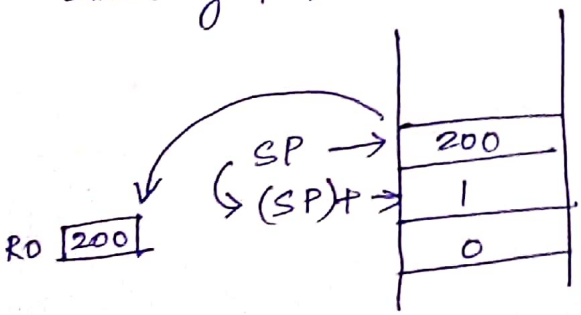


MOVE R0, -(SP)

SP gets decreased by 1 (4 bytes) and value of R0 gets stored to address pointed by SP

3.

Similarly POP



First SP moves value to the R0 and SP gets updated by increasing one mem location

KOWSIK NANDAGOPAND
ROLL NO 31 CSE S4

3.

Binary of 31 is

$(31)_{10} = (01111)_2$ ↓ carry

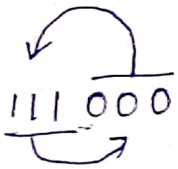
If 8 bit $(00011111)_2 = 31$

LShift #3 R0

Means to shift the contents of R0 to left and put 0 at places. (Shift through carry)

After LShift #3 R0

$(11100)_2$ | 8bit $(11111000)_2$



After Rotate L #3 R0

$(00011)_2$ ↓ carry | 8 bit $(00011111)_2$

In Rotate L through carry assumed zero move byte by byte from LSB to MSB through carry and shift the old MSB to the right

We are actually doing the circular shift operation

4. My phone number ends in 9

∴ option 9 20(R1, R3)

This is a mode of accessing memory. Indexed mode with offset.

ie, [R1 + R3 + 20] - is the memory location
↳ offset

Instruction sequence

1. PCout MARin Read SelectY Add Zin
2. Zout Yin PCin WMFC
3. MDRout IRin
4. R1out Yin
5. R3out, Select Y, Add, Zin
6. Zout Yin
7. Offset-of-IRout Select Y Add Zin // Adds offset 20
8. Zout, R1 end

... (shift through carry)

Kowsik ND

1. Last 4 digits of phone number 3609

$(3609)_{10} = (111000011001)_2$

In 32 bit format,

(1) Little endian. – means that the ending value will be least significant bit ie, index 0 will have LSB of the digit.

31 ... 11 10 9 8 7 6 5 4 3 2 1 0

0 ... 0 0 1 1 1 0 0 0 0 1 1 0 0 1

(2) Big Endian – means that the ending value will be the MSB bit of the binary. ie, index 0 have the MSB of the digit.

31 ... 0

1 0 0 1 1 0 0 0 0 1 1 1 ... 0

~~No single access at zero anymore~~