

Assignment 6

EE15B025 | Ganga Meghanath

Question

Instruction :PUSH : Perform pushing a value to stack. Fetch the operand to be pushed using one of the addressing modes corresponding to your roll number.

Indirect Addressing with Auto pre-increment : Value of 11,12 and 13 should be stored in memory location 49,50 and 51. Now the memory address 50 has to be stored in register B initially. Pass the register B along with the required command.

Answer

Task : Increment the register B (using +1 at register position 6) to point to memory location 51 (initially contains 50), access the value at that memory location (which is 13) and push it to stack (decrement stack pointer before pushing into stack).

MEMORY AND REGISTER VALUES :

Memory	Value
49	11
50	12
51	13

PC	AC	SP	IR	TIR	0	+1	-1	AMASK	SMASK	A	B	C	D	E	F
0	40	69	0	0	0	1	-1	4095	255	0	50	0	0	0	0

MICROINSTRUCTION CODING SHEET :

AMUX 0 = use A latch value

1 = use MBR value

COND 00 = no jump

01 = jump if N=1 (N Flag indicates a negative ALU output)

10 = jump if Z=1 (Z Flag indicates a zero ALU output)

11 = jump always (unconditional jump)

ALU 00 = A + B (add)

01 = A and B (bitwise boolean and)

10 = A (pass the left input straight on through the ALU)

11 = inv(A) (bitwise boolean not of the left input)

SH 00 = no shift (just pass the input on through)

01 = shift right 1 bit

10 = shift left 1 bit

11 = (not used)

MBR 0 = do not load the memory buffer register (MBR)

1 = load the MBR register from the C bus

MAR 0 = do not load the memory address register (MAR)

1 = load the MAR using the 12 low order bits from B bus

RD 0 = do not assert the read signal

1 = assert the read signal

WR 0 = do not assert the write signal

1 = assert the write signal

ENC 0 = do not assert the "enable C" signal

1 = assert the "enable C" signal (to load a register from the C bus)

C 4 bit number specifying the register to be loaded from the C bus

B 4 bit number specifying the register whose contents are to be gated onto the B bus

A 4 bit number specifying the register whose contents are to be gated onto the A bus

ADDR 8 bit number specifying a microprogram address to jump to (if a jump is called for, as shown in the COND field and based on the N Flag and Z Flag values)

	A M U X	C O N D	A L U	S H	M B R	M A R	R D	W R	E N C	C	B	A	ADDR
0	0	00	00	00	0	0	0	0	1	1011	0110	1011	00000000
1	0	00	10	00	0	1	1	0	0	0000	1011	0000	00000000
2	0	00	10	00	0	0	1	0	0	0000	0000	0000	00000000
3	0	00	00	00	0	0	0	0	1	0010	0111	0010	00000000
4	0	00	10	00	0	1	0	1	0	0000	0010	0000	00000000
5	0	00	10	00	0	0	0	1	0	0000	0000	0000	00000000

1 2 2 2 1 1 1 1 1 4 4 4 8

Number of bits in each field

MAL Form of the Microinstructions : Tanenbaum notation

0:	<code>b := b+1;</code>	The register value is incremented and saved
1:	<code>mar := b; rd;</code>	The address is transferred to MAR and read command initiated
2:	<code>rd;</code>	Wait for memory read to complete
3:	<code>sp := sp + -1;</code>	Stack pointer is decremented
4:	<code>mar := sp; wr;</code>	The address pointed to by the stack pointer is sent to MAR
5:	<code>wr;</code>	Wait for memory write